

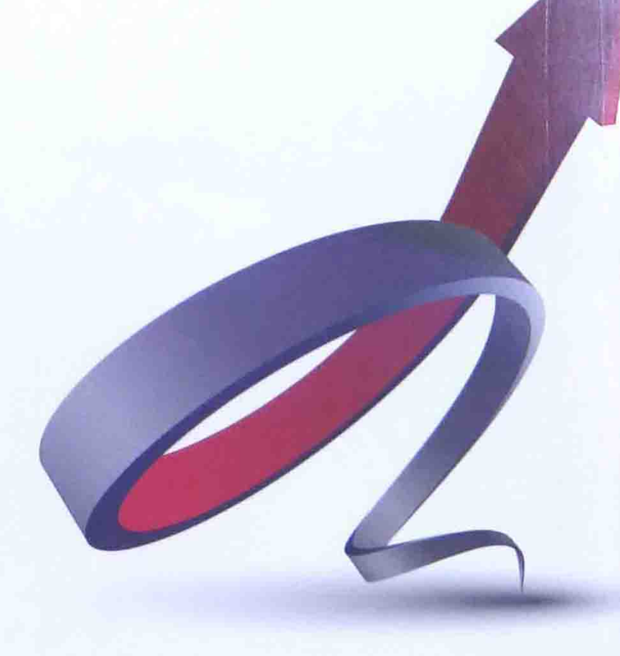
> 华为ICT认证系列丛书

华为技术有限公司与泰克网络实验室 联合创作

华为技术认证

HCNA网络技术实验指南

华为技术有限公司 主编





人民邮电出版社
POSTS & TELECOM PRESS



华为技术认证

HCNA网络技术实验指南

华为技术有限公司 主编



人民邮电出版社
北京

图书在版编目（CIP）数据

HCNA网络技术实验指南 / 华为技术有限公司主编
. -- 北京 : 人民邮电出版社, 2014.5
（华为ICT认证系列丛书）
ISBN 978-7-115-34319-2

Ⅰ. ①H… Ⅱ. ①华… Ⅲ. ①企业—计算机网络
Ⅳ. ①TP393.18

中国版本图书馆CIP数据核字(2014)第001700号

内容提要

为帮助广大ICT从业人员更好地学习信息和网络技术，华为技术有限公司在2012年9月发布了业界首款免费的企业网络仿真软件平台eNSP（Enterprise Network Simulation Platform）。软件平台推出至今，下载量已超过百万，迅速成为ICT从业人员学习信息和网络技术的首选工具。随着学习的深入，越来越多的用户希望能看到与eNSP配套的实验学习指南，从而更好地利用eNSP学习信息和网络技术，并参加华为认证考试。

为此，华为技术有限公司与泰克网络实验室（华为授权培训合作伙伴）联合编写了本书。本书最大的特点是依据HCNA的知识点，基于eNSP搭建企业网络真实场景，并给出了大量的配置案例，将真实场景与配置实例紧密结合，使读者能够更快速、更直观、更深刻地掌握HCNA所需的知识，提高操作技能，增强实战经验。本书主要内容包括eNSP使用说明、VRP基础操作、二层交换技术和三层路由技术等，特别适合于正在学习HCNA或者想进一步提高对网络知识的理解及实际操作技能的读者。

◆ 主　　编　华为技术有限公司
　责任编辑　李　静
　责任印制　杨林杰
◆ 人民邮电出版社出版发行　　北京市丰台区成寿寺路11号
　邮编　100164　　电子邮件　315@ptpress.com.cn
　网址　http://www.ptpress.com.cn
　北京隆昌伟业印刷有限公司印刷
◆ 开本：787×1092　1/16
　印张：27.5　　　　　　2014年5月第1版
　字数：631千字　　　　2014年5月北京第1次印刷

定价：69.00元

读者服务热线：(010)81055488　印装质量热线：(010)81055316
反盗版热线：(010)81055315

华为认证简介

华为认证是华为技术有限公司凭借多年信息通信技术人才培养经验，以及对行业发展的深刻理解，基于 ICT（InformationCommunicationTechnology，信息通信技术）产业链人才个人职业发展生命周期，搭载华为“云—管—端”融合技术，推出的覆盖 IP、IT、CT 以及 ICT 融合技术领域的认证体系，是业界唯一的 ICT 全技术领域认证体系。

华为技术有限公司经过 20 多年在 ICT 行业培训和认证领域的积累，已经在全球形成了完整的培训认证体系，包括自有的培训中心、授权的培训中心以及与高校合作的教育项目，累计参加华为培训的人次已超过 300 万，培训与考试服务覆盖 160 多个国家，平均每天有逾 250 名学员在全球各地接受华为的技术培训。

对行业不同领域的人才，华为均有与之匹配的知识和技能培养解决方案，对其进行准确合理的能力评估。针对个人的职业发展历程，华为提供从工程师到资深工程师、到专家、到架构师，从单一的技术领域到 ICT 融合的职业认证；针对华为的合作伙伴，基于不同岗位，提供销售专家、解决方案专家、售后专家等专业认证。

要全面了解华为认证培训相关信息，请访问华为培训认证主页（http://support.huawei.com/learning）。

要了解华为认证最新动态，请关注华为认证官方微博（http://e.weibo.com/hwcertification）。

通过华为官方论坛链接（http://support.huawei.com/ecommunity/bbs）点击进入华为认证版块（华为职业认证包含的内容如图 1 所示），可以和广大用户一起进行技术问题的探讨，以及考试学习资料的分享。

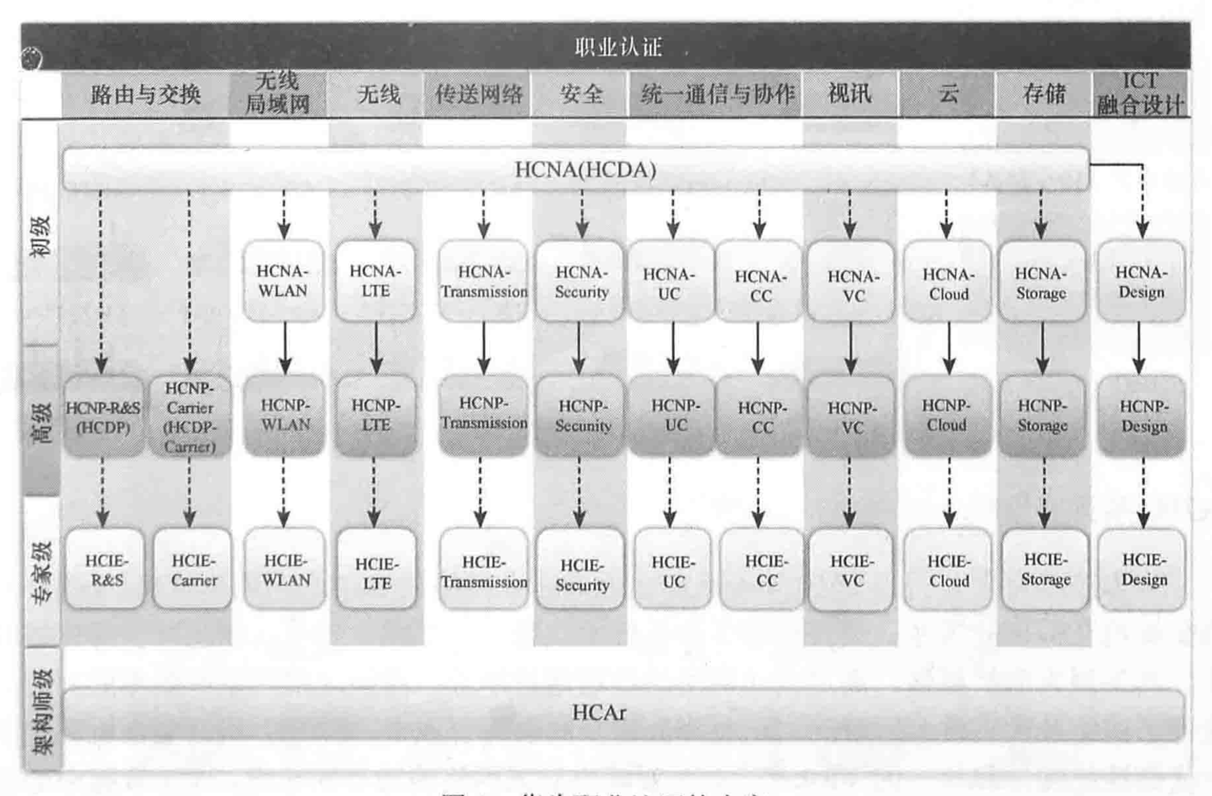


图 1　华为职业认证的内容

华为路由交换产品介绍

AR 系列路由器

2011 年，华为技术有限公司发布了第三代企业接入路由器 AR G3 系列。该系列采用多核 CPU 及大容量交换网，是集安全、语音、互联、无线于一体的多业务的企业路由器，并通过了北美权威机构的评测，性能是业界水平的 2 倍以上，从根本上为企业多业务环境的优质体验提供了保证。

AR G3 系列企业路由器一般位于企业内部与外部网络的连接处，是内部与外部网络之间数据流的唯一出入口，能将多种业务部署在同一设备上，极大地降低了企业网络建设的初期投资与长期运维成本。用户可以根据企业用户规模选择不同规格的 AR G3 路由器作为出口网关设备。AR 系列路由器如图 2 所示。

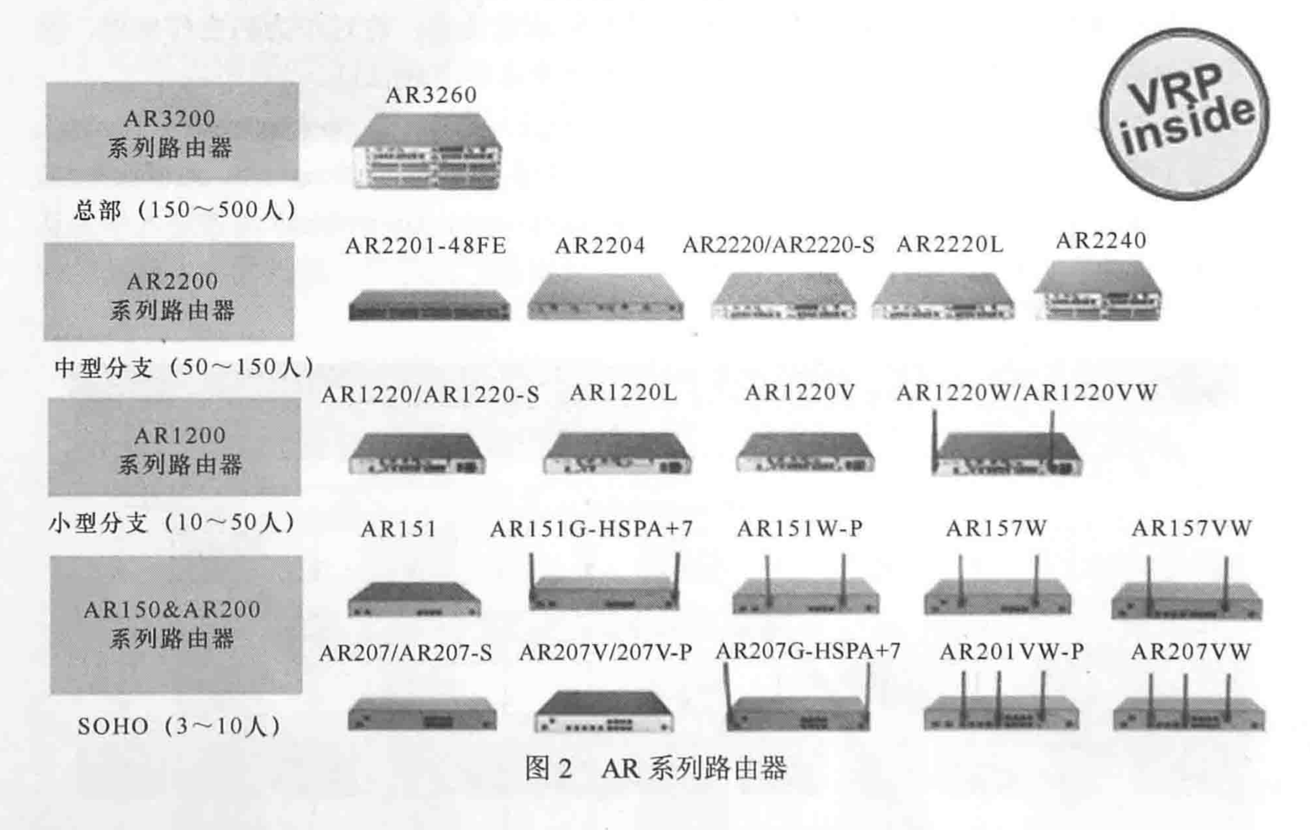


图 2 AR 系列路由器

Sx700 系列交换机

华为技术有限公司于 2010 年 6 月推出面向企业网络的 Sx700 系列交换机。Sx700 系列交换机在提供高带宽、高性能服务的基础上，融合了可靠、安全、绿色环保等先进技术，具备强大的扩展性，满足企业网络的持续演进需求。在提高用户生产效率的同时，保证了网络最大正常运行时间，从而降低用户的总拥有成本。Sx700 系列交换机基于新一代高性能硬件和统一的 VRP 平台，主要解决局域网络的部署和建设，以及数据中心的接入应用。在资质和认证方面，华为交换机在基本功能、节能减排、可靠性、互通性等

了北美权威评测机构的全方位测试和认证，是业界不可多得的高性价比网络产品。Sx700 系列交换机如图 3 所示。

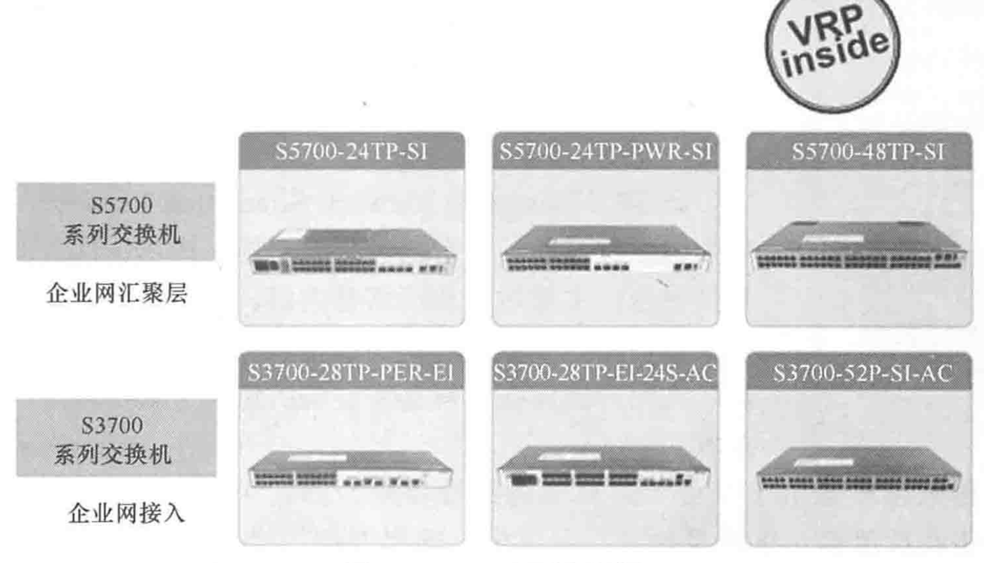


图 3 Sx700 系列交换机

企业网络解决方案

企业网络的构建可根据企业本身的规模大小选择不同的解决方案。例如，一个小型分支机构，可使用 S3700 作为接入层交换机直接连接到作为网络出口的 AR 系列路由器。大中型企业网络通常需要分层设计，接入层可部署 S3700 交换机，实现对不同类型用户终端的接入；汇聚层可采用 S5700 交换机，下行通过千兆接口连接接入交换机，上行通过万兆光口连接核心层路由器；核心层可根据需求选择不同规格的 AR 路由器作为核心层设备。

华为数通产品企业网络解决方案场景如图 4 所示。

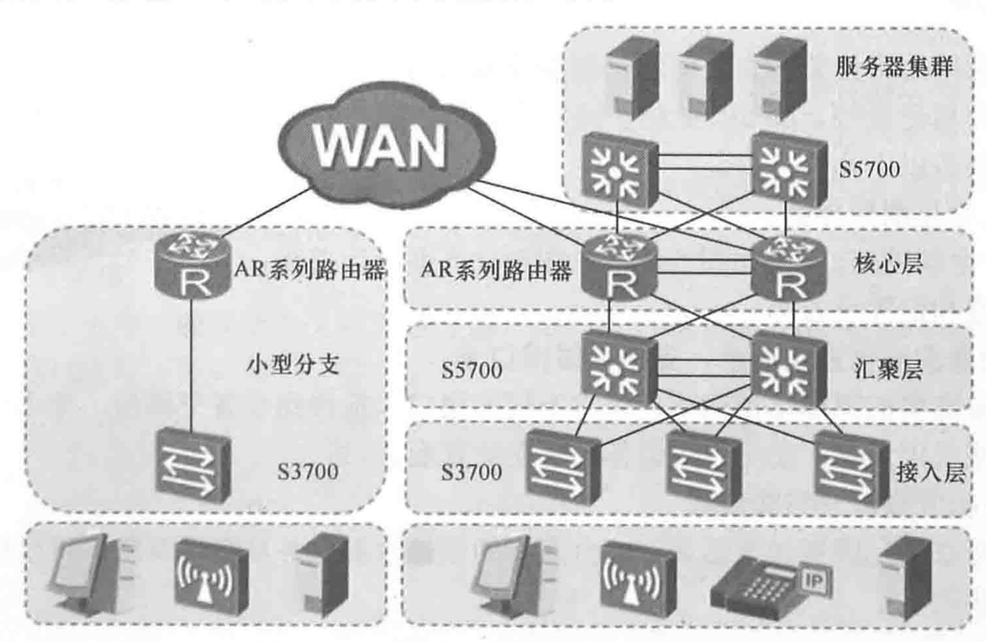


图 4 华为数通产品企业网络解决方案场景

eNSP 介绍

eNSP 简介



eNSP（Enterprise Network Simulation Platform）是一款由华为自主开发的、免费的、可扩展的、图形化操作的网络仿真工具平台，主要对企业网络路由器、交换机及相关物理设备进行软件仿真，完美呈现真实设备实景，支持大型网络模拟，可让广大用户能够在没有真实设备的情况下模拟演练，学习网络技术。

针对越来越多的 ICT 从业者对真实网络设备模拟的需求，eNSP 企业网络仿真平台拥有着仿真程度高、更新及时、界面友好、操作方便等特点。这款仿真软件运行的是与真实设备同样的 VRP 操作系统，能够最大程度地模拟真实设备环境。用户可以利用 eNSP 模拟工程开局与网络测试，高效地构建企业优质的 ICT 网络。eNSP 支持与真实设备对接，以及数据包的实时抓取，可以帮助用户深刻理解网络协议的运行原理，协助进行网络技术的钻研和探索。另外，用户还可以利用 eNSP 模拟华为认证相关实验（HCNA、HCNP-R&S、HCIE-R&S 等），能更轻松地获得华为认证，成就技术专家之路。

eNSP 的免费发布，将为用户提供近距离体验华为设备的机会。无论是操作数通产品、维护现网的技术工程师，还是教授网络技术的培训讲师，或者是想要获得华为认证，获得能力认可的在校学生，相信都可以从 eNSP 中受益。

eNSP 的特点

针对影响用户体验的主要问题，例如安装是否方便、仿真度是否够高、是否可视化操作、是否可更新等，eNSP 做到了扬各家之长，避各家之短，给用户带来极佳的操作体验，它具备以下几个特点。

1．人性化图形界面

eNSP 全新的 UI 界面如图 5 所示。图形化界面不但美观，且操作时可轻松上手，包括拓扑搭建和配置设备等。

2．设备图形化直观展示，支持插拔接口卡

在设备真实的图形化视图下，可将不同的接口卡拖拽到设备空槽位，单击电源开关即可启动或关闭设备，使用户对设备的感受更直观。

3．多机互联，分布式部署

最多可在 4 台服务器上部署 200 台左右的模拟设备，并且实现互联，可以模拟大型复杂网络实验。

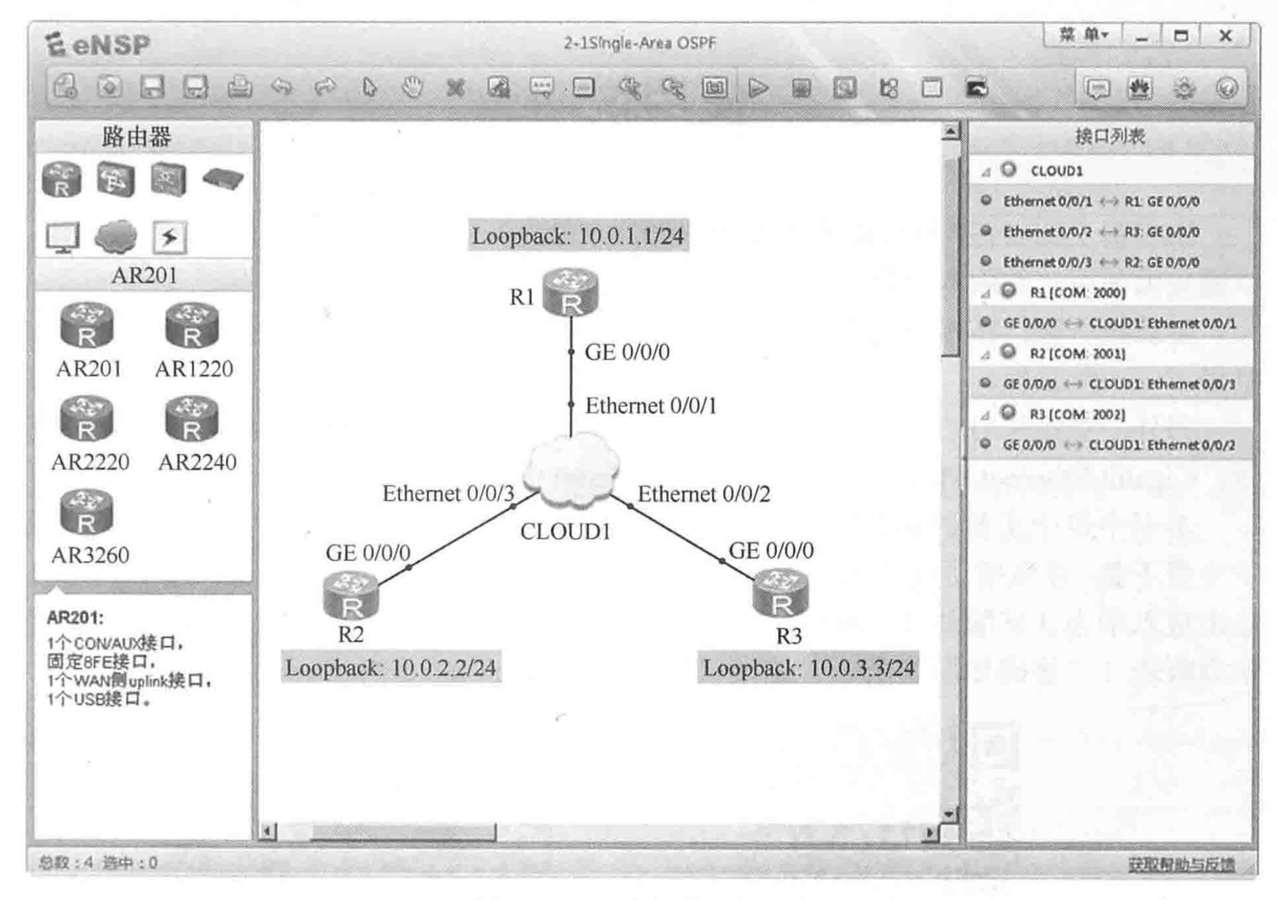


图 5　eNSP 全新的 UI 界面

4．高度仿真，实景再现，支持设备功能多

高度仿真的二层转发，运行华为通用路由平台 VRP 系统，支持对路由器、交换机各种特性的仿真和模拟（包括 STP/RSTP/MSTP、Mux VLAN、SEP、GVRP 等各种协议），提供 AR 全系列仿真款型，支持 NAT、防火墙、IPSec、SSLVPN、MQC、AC 等功能。

5．不断增加的功能特性模拟

随着真实产品的升级更新，软件将支持增加更多更新的功能特性与之对应。用户发现任何问题都可以通过华为 support-e 官网论坛进行问题反馈，华为技术有限公司有专人负责技术支持和疑问解答，亦可通过邮箱 eNetwork_tools@huawei.com 进行反馈，反馈的问题和建议将通过后续月度版本予以快速响应。

6．完全免费

软件完全免费，面向所有人群开放下载，用户可登录 http://enterprise.huawei.com/cn/ 华为官方网站进行下载。

使用说明

本书基于的 eNSP 软件版本为 V100R002C00B210，请读者于官网上对应下载使用，以避免因使用软件版本不符而造成实际实验操作与书中内容不一致。

受篇幅所限，在本书的实验现象输出命令中，凡是不与实验主题相关的部分都以省略号“……”替代。

另外，实验常用的设备接口在正文中以缩略词表示，如 Ethernet 0/0/0 以 E 0/0/0 表示；Gigabit Ethernet 0/0/0 以 GE 0/0/0 表示；Serial 0/0/0 以 S 0/0/0 表示。

本书中每个实验的拓扑图、思考题答案及最终配置都以电子文件形式在网页上提供免费下载，详情请访问华为官方论坛链接（http://support.huawei.com/ecommunity/bbs）点击进入华为认证版块（二维码如图 6 所示）；或者访问泰克网络实验室官方网站中的华为版块（二维码如图 7 所示）链接（http://www.tech-lab.cn/huawei）。



图 6 华为官方论坛中华为论证版块中的二维码



图 7 泰克网络实验室官方网站中华为版块中的二维码

本书常用图标



序

作为全球领先的信息与通信解决方案供应商，华为以“丰富人们的沟通和生活”为愿景，利用在ICT领域的专业技术和经验，帮助不同地区的人们平等、自由地接入到信息社会，确保所有人都能享受到信息和通信服务的基本权利，致力于消除数字鸿沟。我们提倡和致力于信息和通信技术的普及，增加教育机会并培养ICT人才。

为帮助广大ICT从业人员更好地学习信息和网络技术，华为技术有限公司在2012年9月发布了业界首款免费的企业网络仿真软件平台eNSP（Enterprise Network Simulation Platform）。这款仿真软件平台主要对企业网络路由器、交换机、防火墙、WLAN等网络设备进行软件仿真，具备高仿真、界面友好、操作方便、版本更新及时等特点。eNSP一经推出就受到社会的广泛关注和欢迎，下载量已超过百万，迅速成为ICT从业人员学习信息和网络技术的首选工具。随着学习的深入，越来越多的ICT从业人员希望能看到与eNSP配套的实验学习指南，从而更好地利用eNSP学习信息和网络技术，并参加华为认证考试。

为满足广大ICT从业者的学习需求，我们联合华为授权培训合作伙伴——泰克网络实验室，组织多名经验丰富的老师编写了本实验指南。对于一个ICT从业人员来说，理论知识固然重要，但动手实操能力更不可少。许多仅仅从书本上学习信息和网络技术原理的ICT从业人员，在面对真实的组网时往往会无所适从。本书最大的特点是依据HCNA的知识点，基于eNSP搭建企业网络真实场景，配置案例系统丰富，实验步骤细致严谨，文字描述详实准确。

这是“华为ICT认证系列丛书”中一本不可多得的实验学习指南，希望它能够帮助广大读者不断积累实战经验，同时系统地梳理和巩固书本上所学的技术理论知识，使得理论与实战相得益彰，从而进一步帮助读者成为ICT领域的专家人才，在ICT行业大展身手！

全球培训与认证部部长

华为企业业务集团

2014年4月

前　言

本书实验包括：“原理概述”、“实验目的”、“实验内容”、“实验拓扑”、“实验步骤”、“思考”等模块。读者应首先阅读“原理概述”和“实验目的”，了解本实验应该掌握的知识和技能，再进行实验操作。实验过程中请读者仔细阅读“实验步骤”中的说明，这些内容将很好地展示实验的思路。最后的“思考”模块，可以启发读者进行进一步思考，使读者能更加深刻地理解相关知识。

为了更便于读者学习和练习，我们把每个实验项目都做成了独立的 eNSP 实验软件包，包括每个实验的最终配置和思考题答案等内容都放到网站上提供给读者下载和学习，读者可以在本书的“使用说明”中找到这些网址。

适用读者对象

本书的基本定位是华为 HCNA 认证的参考书，全方位涵盖了 HCNA 的知识点。本书适合于以下几类读者。

- 使用华为路由器和交换机的用户

本书可帮助用户更加熟练地操作和使用华为设备，加深对网络技术的理解，通过实验模拟现网，增加实际项目经验。

- ICT 从业人员

本书可作为工具用书，帮助 ICT 从业人员熟悉华为设备，具备快速配置华为路由器和交换机的能力，掌握相关网络技术的应用。本书更有助于 ICT 从业人员获取华为认证，提升在企业中的个人价值。

- 高校学生

本书可作为华为信息与网络技术学院的实验教材，也可作为计算机通信等相关专业学生的自学参考书。配合 eNSP 软件，本书可以帮助学生快速地熟悉华为网络设备的操作，理解和掌握信息和网络技术，使学生能更快地积累企业网络实践经验，更早地获得华为认证，在今后的职业生涯中有一个更好的起步。

- 对信息和网络技术感兴趣的爱好者

本书可作为学习信息和网络技术的参考书籍，使爱好者了解华为产品和技术的特点，掌握华为产品和技术的应用，并为其进一步的技术研究提供工具和指导。

本书主要内容

全书共分为 14 章，所有实验都是基于 eNSP 作为实验工具，按照 HCNA 的知识点进行设计。

第 1 章：eNSP 及 VRP 基础操作

本章介绍了 eNSP 软件的基本操作方法、华为 VRP 通用路由平台的基本操作方法、

IP 基础配置、Telnet、Stelnet、FTP 的操作实例。

第 2 章：交换机基础配置

本章介绍了华为交换机的基本配置方法，给出了 ARP、ARP proxy 的配置实例。

第 3 章：VLAN

本章给出了 VLAN 的 Access 接口、Trunk 接口、Hybird 接口的配置实例，以及分别通过单臂路由和三层交换机实现 VLAN 间通信的配置实例。

第 4 章：生成树

本章给出了 STP 的配置实例，并详细介绍了 STP 的选举规则和定时器，给出了 RSTP 和 MSTP 的基础配置实例。

第 5 章：其他交换技术

本章给出了 GVRP、Smart-Link 与 Monitor-Link、Eth-trunk 的配置实例。

第 6 章：静态路由

本章给出了静态路由和默认路由的配置实例，并在此基础上给出了浮动静态路由和负载均衡的配置实例。

第 7 章：RIP

本章以 RIP 路由协议为主题，给出了包括基本配置、认证、汇总、版本兼容、故障排除等多方面特性的配置实例。

第 8 章：OSPF

本章以 OSPF 路由协议为主题，给出了包括单区域配置、多区域配置、认证、被动接口等多方面特性的配置实例。

第 9 章：VRRP

本章给出了 VRRP 的基本配置、多备份组、跟踪接口及认证的配置实例。

第 10 章：基础过滤工具

本章给出了常用过滤工具、基本的访问控制列表、高级的访问控制列表及前缀列表的配置实例。

第 11 章：广域网

本章介绍了 HDLC 和 PPP 的基础配置方法，给出了 PAP 和 CHAP 认证的配置实例，介绍了帧中继的基础配置方法，给出了帧中继网络中 OSPF 协议的配置实例。

第 12 章：DHCP

本章给出了基础全局地址池和基础接口地址池的 DHCP 配置实例，同时也给出了 DHCP 中继的配置实例。

第 13 章：IPv6

本章介绍了 IPv6 地址的基础配置，给出了 RIPng 路由协议和 OSPFv3 路由协议的配置实例。

第 14 章：其他特性

本章介绍了 SNMP 协议的基础配置，给出了 GRE 协议的配置实例，给出了使用 eNSP 软件与真实 PC 进行桥接的方法示例，以及 NAT 技术的配置实例。

鸣谢

本书由华为技术有限公司与泰克网络实验室联合编写，经过双方多位编委老师半年多时间的严格审校、修改和完善，这本实验指南终于高质量完成并得以顺利地出版上市。在此感谢泰克网络实验室各位老师的付出和大力支持，感谢人民邮电出版社各位编辑老师，以及各位编委的辛勤工作！以下是本书主要参与编写老师的介绍。

徐一鸣，具有多年 ICT 从业经验，现任华为路由交换认证架构师，华为 HCIE R&S 项目负责人，负责设计和开发华为路由交换领域的认证培训教材，以及培训授课工作；同时负责 HCIE R&S 考试的执行及面试等工作。

刘晶晶，具有多年 ICT 从业经验，华为路由交换资深讲师，华为路由交换讲师认证主考官，负责开发华为路由交换领域的认证培训教材，以及培训授课工作。

林康平，全球第一个 HCIE 的诞生地——泰克网络实验室的创始人，多家大中企业的资深技术顾问。负责过多项金融行业的 ICT 网络优化、互联网服务提供商三网合一等大型项目。他希望通过自己的技术和经验培养更多优秀的 ICT 行业人才，其座右铭是：我们一直在努力，从为自己更优秀到帮助别人更优秀。

参与本书编写和审稿的老师虽然有多年 ICT 从业经验，但因时间仓促，错漏之处在所难免，望读者不吝赐教，在此表示衷心的感谢。读者对于本书有任何意见和建议可以发送邮件至 Learning@huawei.com。

以下是参与本书编写和技术审校人员的名单。（排名不分先后）

主　　编：涂文杰

编委人员：刘晶晶、徐一鸣、宋新华、林康平、邵　岩、陈　亮、陈世杰、刘大伟、丁林琦、彭　丹、沈义鸿

技术审校：江永红、刘　洋、张　超、李云超

目　　录

第 1 章　eNSP 及 VRP 基础操作……0

1.1　认识 eNSP……2
1.2　熟悉 VRP 基本操作……13
1.3　熟悉常用的 IP 相关命令……18
1.4　配置通过 Telnet 登录系统……25
1.5　配置通过 STelnet 登录系统……29
1.6　配置通过 FTP 进行文件操作……34

第 2 章　交换机基础配置……40

2.1　交换机基础配置……42
2.2　理解 ARP 及 Proxy ARP……44

第 3 章　VLAN……54

3.1　VLAN 基础配置及 Access 接口……56
3.2　配置 Trunk 接口……60
3.3　理解 Hybrid 接口的应用……65
3.4　利用单臂路由实现 VLAN 间路由……75
3.5　利用三层交换机实现 VLAN 间路由……81

第 4 章　生成树……86

4.1　STP 配置和选路规则……88
4.2　配置 STP 定时器……94
4.3　RSTP 基础配置……101
4.4　MSTP 基础配置……110

第 5 章　其他交换技术……120

5.1　GVRP 基础配置……122
5.2　Smart Link 与 Monitor Link……129
5.3　配置 Eth-Trunk 链路聚合……134

第 6 章 静态路由 ······ 142

6.1 静态路由及默认路由基本配置 ······ 144
6.2 浮动静态路由及负载均衡 ······ 153

第 7 章 RIP ······ 162

7.1 RIP 路由协议基本配置 ······ 164
7.2 配置 RIPv2 的认证 ······ 168
7.3 RIP 路由协议的汇总 ······ 175
7.4 配置 RIP 的版本兼容、定时器及协议优先级 ······ 181
7.5 配置 RIP 抑制接口及单播更新 ······ 187
7.6 RIP 与不连续子网 ······ 195
7.7 RIP 的水平分割及触发更新 ······ 204
7.8 配置 RIP 路由附加度量值 ······ 209
7.9 RIP 的故障处理 ······ 214
7.10 RIP 的路由引入 ······ 226

第 8 章 OSPF ······ 234

8.1 OSPF 单区域配置 ······ 236
8.2 OSPF 多区域配置 ······ 240
8.3 配置 OSPF 的认证 ······ 246
8.4 OSPF 被动接口配置 ······ 252
8.5 理解 OSPF Router-ID ······ 257
8.6 OSPF 的 DR 与 BDR ······ 263
8.7 OSPF 开销值、协议优先级及计时器的修改 ······ 269
8.8 连接 RIP 与 OSPF 网络 ······ 275
8.9 使用 RIP、OSPF 发布默认路由 ······ 279

第 9 章 VRRP ······ 284

9.1 VRRP 基本配置 ······ 286
9.2 配置 VRRP 多备份组 ······ 291
9.3 配置 VRRP 的跟踪接口及认证 ······ 296

第 10 章 基础过滤工具 ······ 302

10.1 配置基本的访问控制列表 ······ 304
10.2 配置高级的访问控制列表 ······ 308
10.3 配置前缀列表 ······ 312

第 11 章　广域网 ······ 320

11.1　WAN 接入配置 ······ 322
11.2　PPP 的认证 ······ 325
11.3　帧中继基本配置 ······ 332
11.4　OSPF 在帧中继网络中的配置 ······ 337

第 12 章　DHCP ······ 346

12.1　配置基于接口地址池的 DHCP ······ 348
12.2　配置基于全局地址池的 DHCP ······ 352
12.3　配置 DHCP 中继 ······ 358

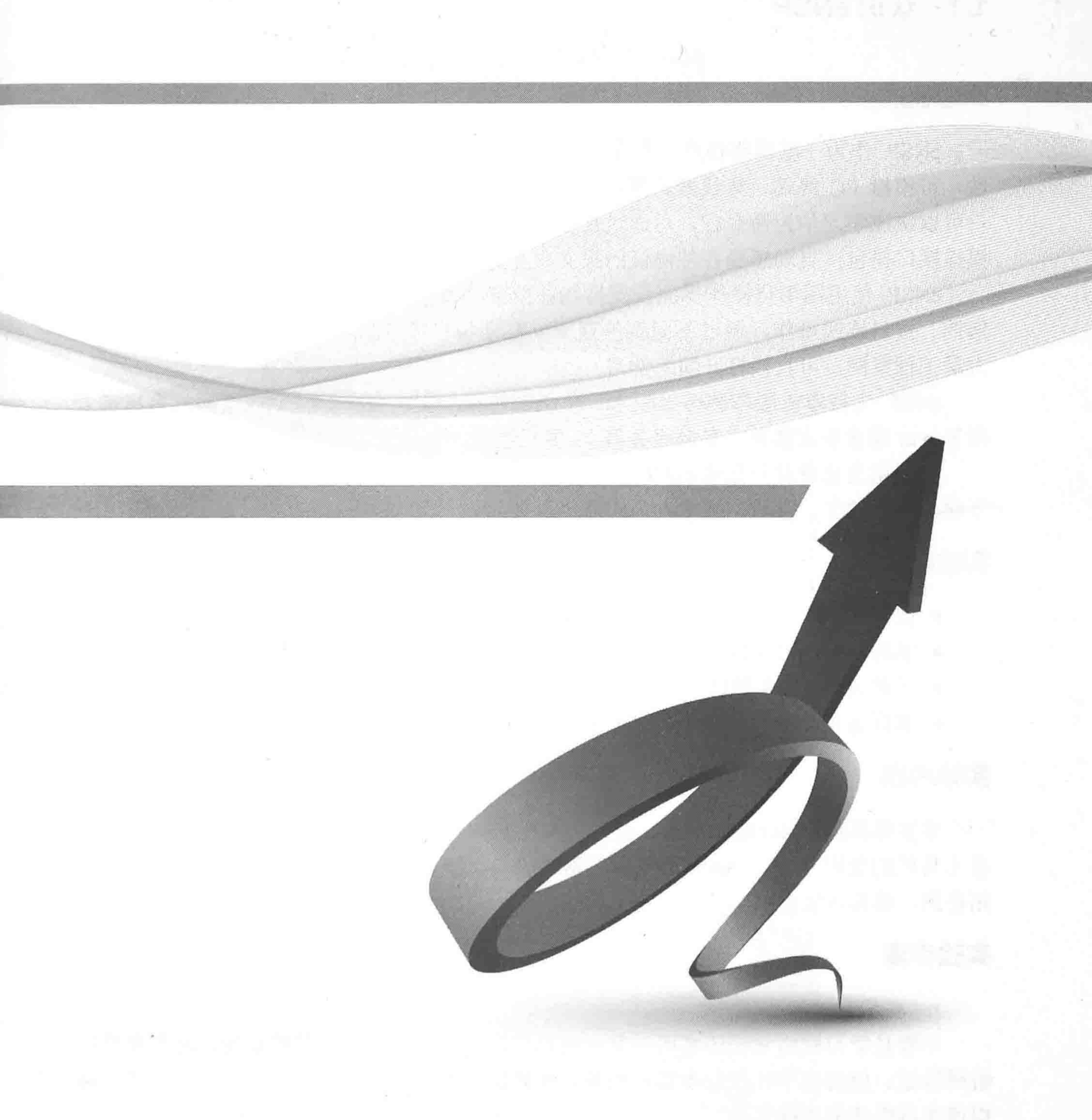
第 13 章　IPv6 ······ 366

13.1　IPv6 基础配置 ······ 368
13.2　RIPng 基础配置 ······ 372
13.3　OSPFv3 基础配置 ······ 377

第 14 章　其他特性 ······ 384

14.1　实现 eNSP 与真实 PC 桥接 ······ 386
14.2　SNMP 基础配置 ······ 391
14.3　GRE 协议基础配置 ······ 395
14.4　配置 NAT ······ 401

附录　命令索引 ······ 410



1.1 认识 eNSP

原理概述

eNSP 作为一款网络仿真工具平台，可模拟华为企业级路由器和交换机的大部分特性，可模拟 PC 终端、集线器、网络云、帧中继交换机等。通过仿真设备配置功能，用户可以快速学习华为命令行，可通过真实网卡实现与真实网络设备的对接，并且还可以模拟接口抓包，直观感受各种协议的报文交互过程。

eNSP 使用图形化操作界面，支持拓扑创建、修改、删除、保存等操作；支持设备拖拽、接口连线操作，通过不同颜色直观反映设备与接口的运行状态。eNSP 还预置了大量工程案例，可直接打开演练学习。

eNSP 支持单机版本和多机版本，单机部署指只在一台主机上完成组网，多机部署指 Server 端分布式部署在多台服务器上。多机组网场景最大可模拟 200 台设备组网规模。

华为完全免费对外开放 eNSP，直接下载安装即可使用，无需申请 license。初学者、专业人员、学生、讲师、技术人员均能免费使用。

实验目的

- 认识 eNSP
- 掌握 eNSP 的安装方法
- 了解 eNSP 的各种功能
- 掌握运用 eNSP 搭建网络拓扑并进行实验的操作方法

实验内容

本实验将从安装 eNSP 开始，全方位介绍 eNSP 的功能，包括 eNSP 软件的主界面、各工具栏的使用方法、eNSP 所支持的网络设备以及如何灵活地使用这些设备搭建网络拓扑图，模拟现实组网。

实验步骤

1. 安装 eNSP

不管是学习网络知识还是用于复现现网问题或项目交付前的预模拟等，都需要模拟组网验证。但现实中往往缺少真实设备，而通过 eNSP 可以很方便地组建虚拟网络，模拟现实网络环境进行实验。

在华为官方网站（http://enterprise.huawei.com）上可以下载到最新版本的 eNSP 安装包。由于 eNSP 上每台虚拟设备都要占用一定的内存资源，所以 eNSP 对系统的最低配置要求为：CPU 双核 2.0GHz 或以上，内存 2GB，空闲磁盘空间 2GB，操作系统为 Windows XP、Windows Server 2003 或 Windows 7，在最低配置的系统环境下组网设备最大数量为 10 台。

安装 eNSP 前请先检查系统配置，确认满足最低配置后再进行安装，步骤如下。

步骤 1：双击安装程序文件，打开安装向导。

步骤 2：在“选择安装语言”对话框中选择“中文（简体）”，单击“确定”按钮，

如图 1-1 所示。

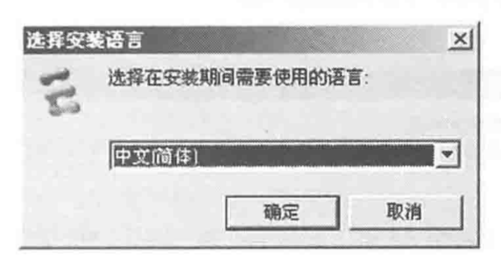


图 1-1　选择安装语言

步骤 3：进入欢迎界面，单击“下一步（N）”按钮，如图 1-2 所示。

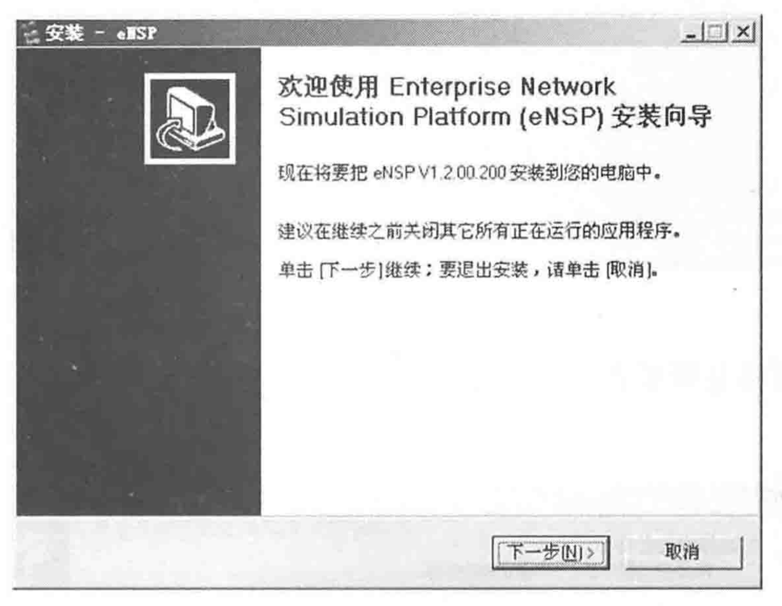


图 1-2　欢迎界面

步骤 4：设置安装的目录（整个目录路径都不能包含非英文字符），单击“下一步（N）”按钮，如图 1-3 所示。

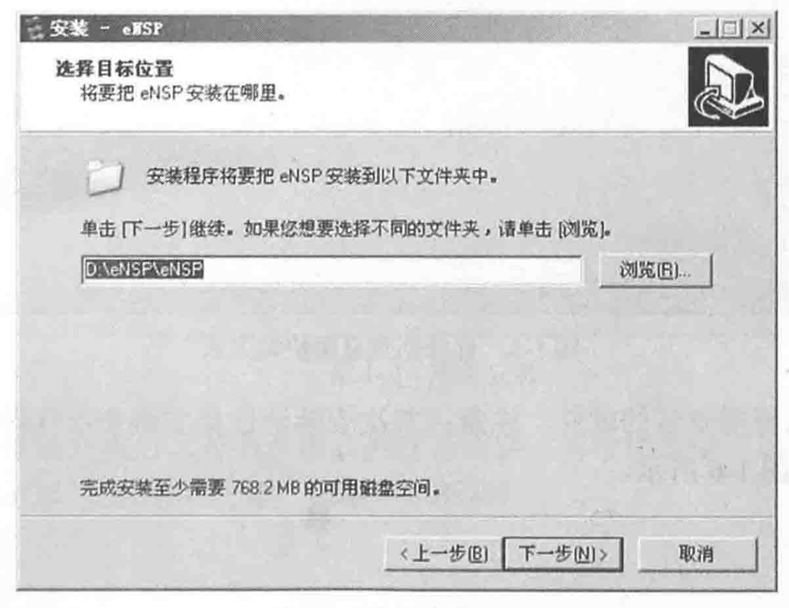


图 1-3　选择安装位置

步骤 5：设置 eNSP 程序快捷方式在开始菜单中显示的名称，单击“下一步（N)”按钮，如图 1-4 所示。

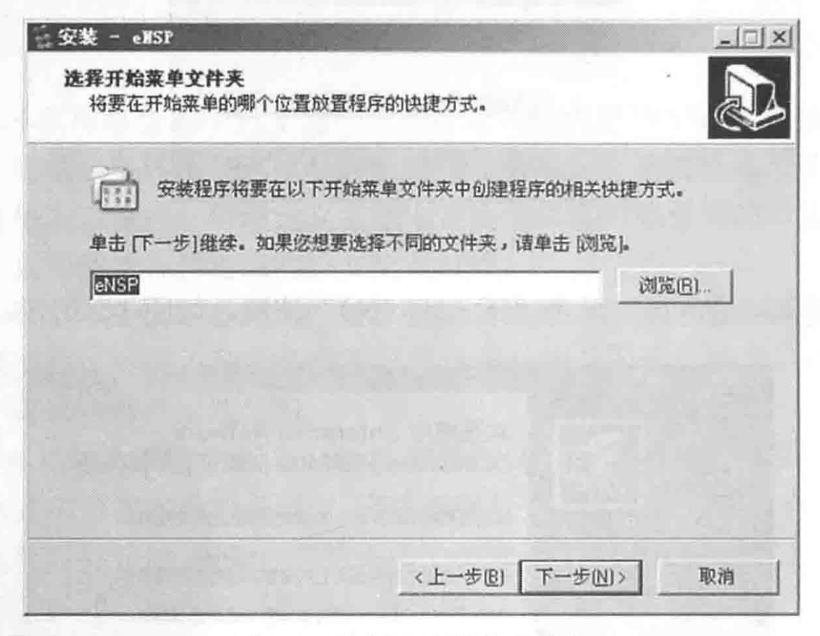


图 1-4　选择开始菜单文件夹

步骤 6：选择是否要在桌面创建快捷方式，单击“下一步（N)”按钮，如图 1-5 所示。

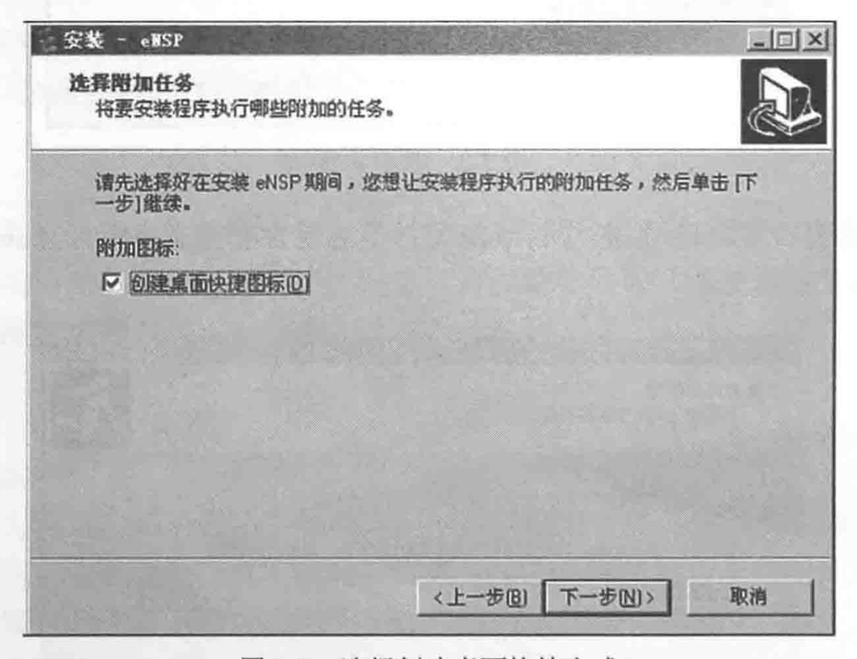


图 1-5　选择创建桌面快捷方式

步骤 7：选择需安装的软件，注意：首次安装请选择安装全部软件，单击“下一步（N)”按钮，如图 1-6 所示。

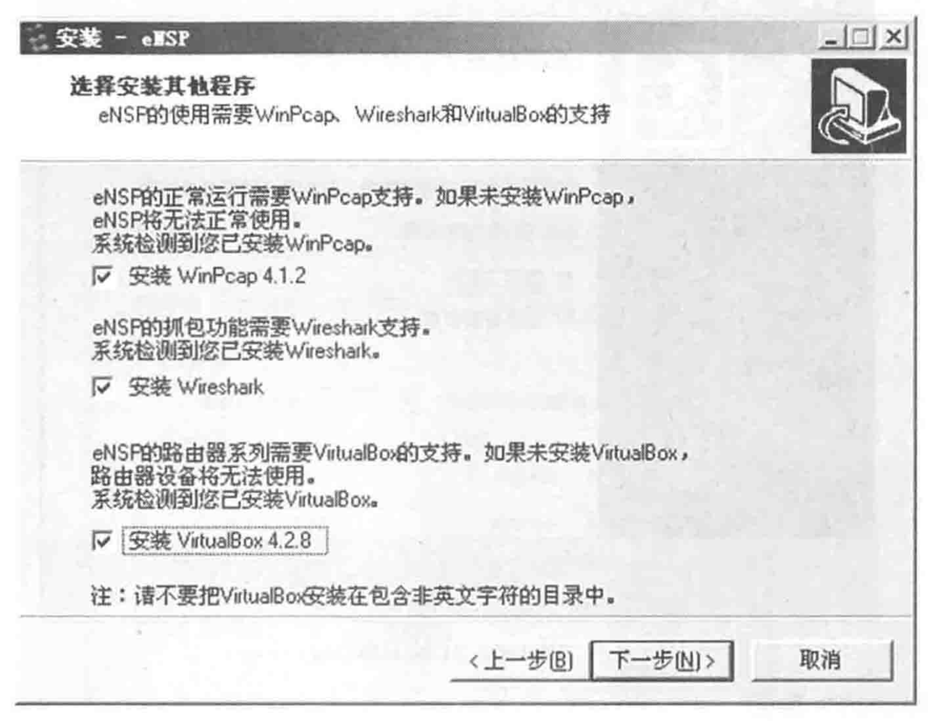


图 1-6 选择安装其他程序

步骤 8：确认安装信息后，单击“安装（I）”按钮开始安装，如图 1-7 所示。

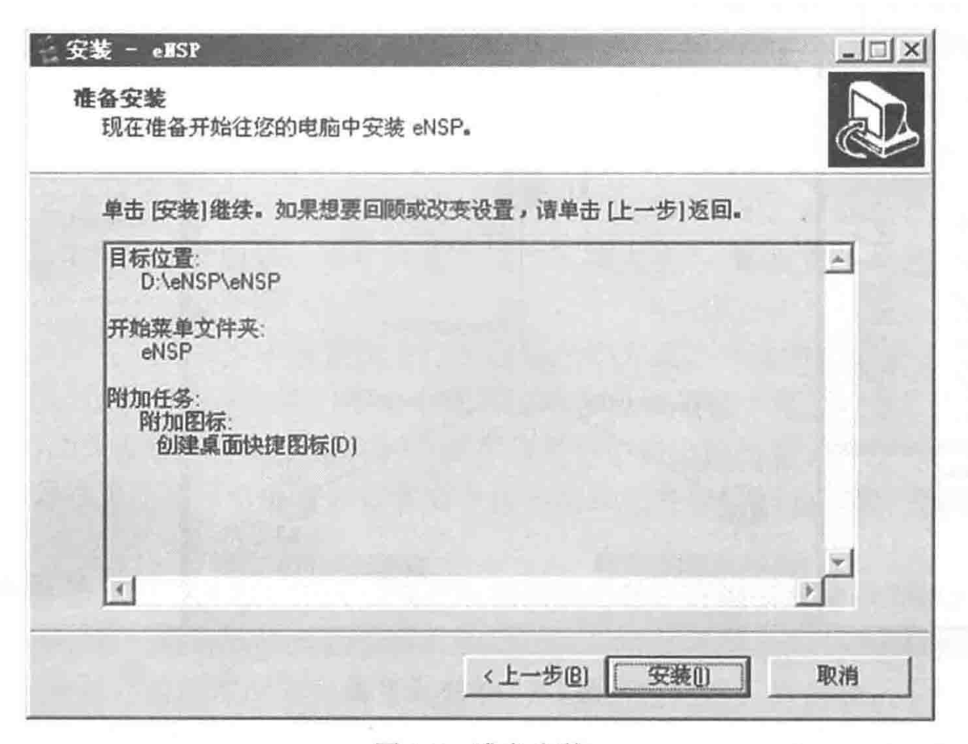


图 1-7 准备安装

步骤 9：安装完成后，若不希望立刻打开程序，可取消选中“运行 eNSP”复选框。单击“完成（F）”按钮结束安装，如图 1-8 所示。

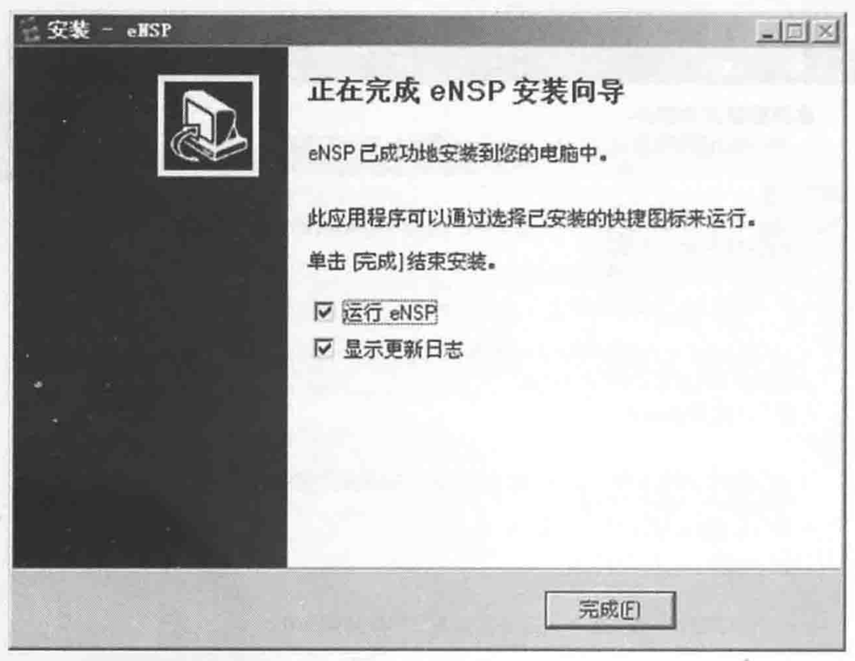


图 1-8 安装完成

2. 熟悉 eNSP 界面

启动 eNSP 模拟器，可以看到其主界面，如图 1-9 所示。

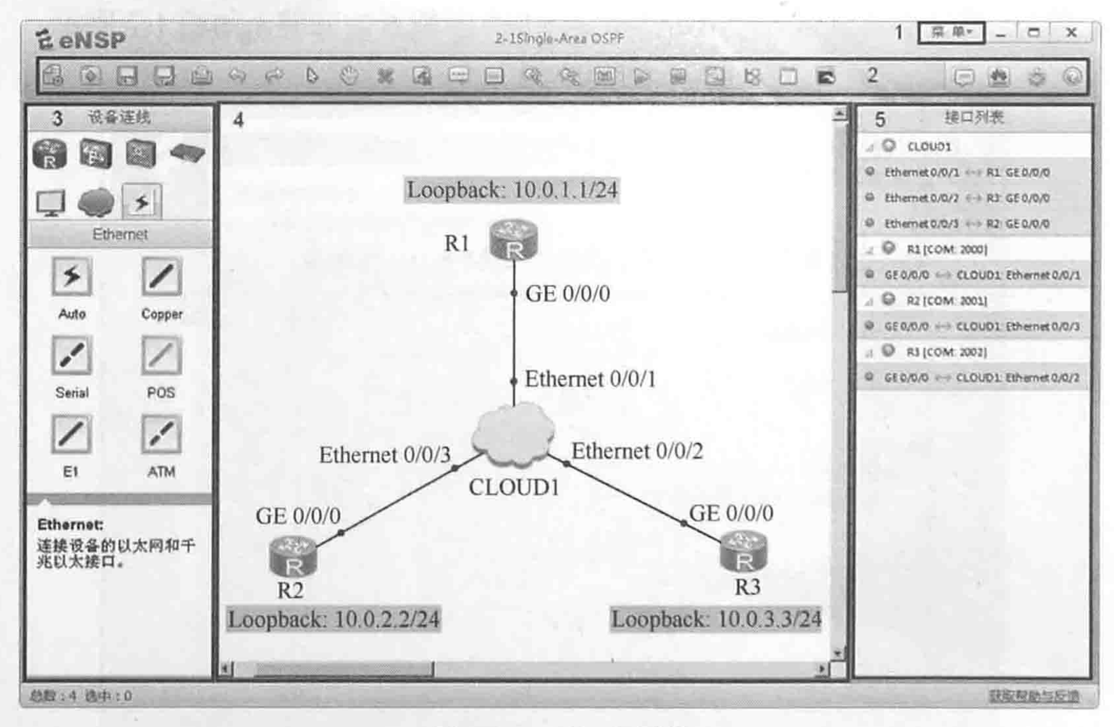


图 1-9 eNSP 主界面

eNSP 主界面分为五大区域。区域 1 是主菜单，提供“文件”、“编辑”、“视图”、“工具”、“帮助”菜单，它们的作用如下。

- “文件”菜单用于拓扑图文件的打开、新建、保存、打印等操作。
- “编辑”菜单用于撤销、恢复、复制、粘贴等操作。
- “视图”菜单用于对拓扑图进行缩放和控制左右侧工具栏区的显示。
- “工具”菜单用于打开调色板工具添加图形、启动或停止设备、进行数据抓包和

各选项的设置。

■ “帮助”菜单用于查看帮助文档、检测是否有可用更新、查看软件版本和版权信息。

■ 进入工具菜单，选择“选项”命令，在弹出的“选项”对话框中设置软件的参数，如图 1-10 所示。



图 1-10　选项

■ 在“界面设置”选项卡中可以设置拓扑中的元素显示效果，比如是否显示设备标签和型号、是否显示背景图等；还可设置“工作区域大小”，即设置工作区的宽度和长度。

■ 在“CLI 设置”选项卡中设置命令行中信息保存方式。当选中“记录日志”时，设置命令行的显示行数和保存位置。当命令行界面内容行数超过“显示行数”中的设置值时，系统将自动保存超过行数的内容到“保存路径”中指定的位置。

■ 在“字体设置”选项卡中可以设置命令行界面和拓扑描述框的字体、字体颜色、背景色等参数。

■ 在“服务器设置”选项卡中可以设置服务器端参数，详细信息请参考帮助文档。

■ 在“工具设置”选项卡中可以指定“引用工具”的具体路径。

区域 2 是工具栏，提供常用的工具（见表 1-1），如新建拓扑、打印等。

表 1-1　工具栏常用图标说明

工具	简要说明	工具	简要说明
	新建拓扑		添加文本
	打开拓扑		调色板，可编辑添加各种图形
	保存拓扑		放大

（续表）

工具	简要说明	工具	简要说明
	另存为		缩小
	打印拓扑		恢复原大小
	撤销上次操作		启动设备
	重复上次操作		停止设备
	恢复鼠标		采集数据报文
	选定工作区，便于移动		显示/隐藏所有接口名称
	删除对象		显示网格
	删除所有连线		打开拓扑中所有路由器和交换机的命令行界面
	华为论坛链接		华为官网链接
	选项设置		帮助文档

在工具栏区域最右边有 4 个按钮，第 1 个是华为论坛的链接按钮，点击后可进入华为官方论坛，进行各种提问和参与讨论；第 2 个是华为官网的链接按钮；第 3 个是“设置”按钮，可进行界面的设置、字体的设置等，与“工具”菜单中的“选项”一致；第 4 个是“帮助文档”按钮，其中详细介绍了当前版本的 eNSP 支持的所有设备特性、各种功能以及如何配置服务器和用户端等。

区域 3 是网络设备区，提供设备和网线，如图 1-11 所示。每种设备都有不同型号，比如点击路由器图标，设备型号区将提供 AR1220、AR2220 等各种路由器，供选择到工作区。

区域 4 是工作区，在此区域可以灵活创建网络拓扑，如图 1-12 所示。

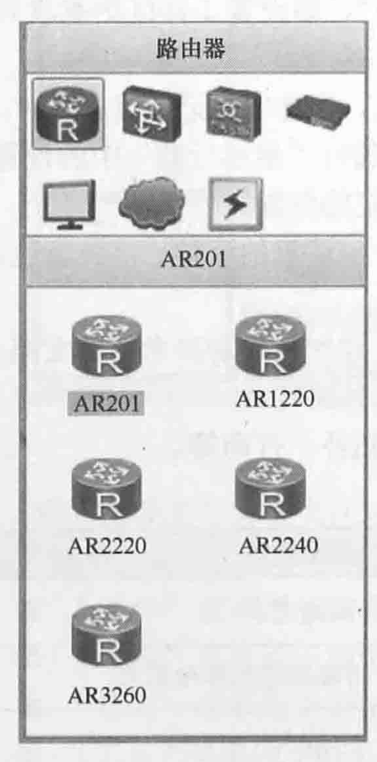


图 1-11 网络设备选择区

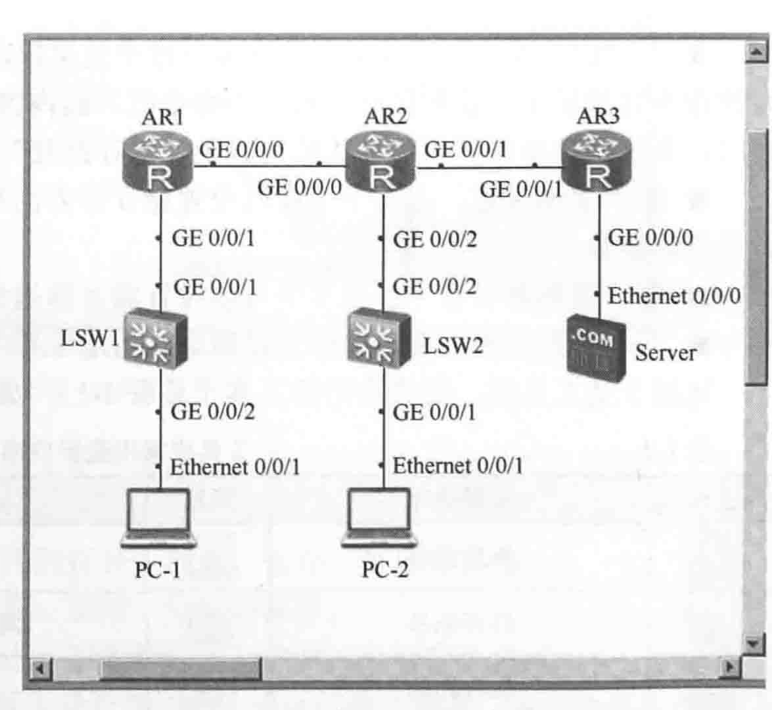


图 1-12 工作区

区域 5 是设备接口区，显示拓扑中的设备和设备已连接的接口，可以通过观察指示灯了解接口运行状态，如图 1-13 所示。浅灰色表示设备未启动或接口处于物理 DOWN 状态；深灰色表示设备已启动或接口处于物理 UP 状态；黑色表示接口正在采集报文。在处于物理 UP 状态的接口名上单击鼠标右键，可启动/停止接口报文采集。

3. 网络设备配置

在 eNSP 中，可以利用图形化界面灵活地搭建需要的拓扑组网图，其步骤如下。

步骤 1：选择设备。主界面左侧为可供选择的网络设备区，将需要的设备直接拖至工作区。每台设备带有默认名称，通过单击可以对其进行修改。还可以使用工具栏中的文本按钮和调色板按钮在拓扑中任意位置添加描述或图形标识，如图 1-14 所示。

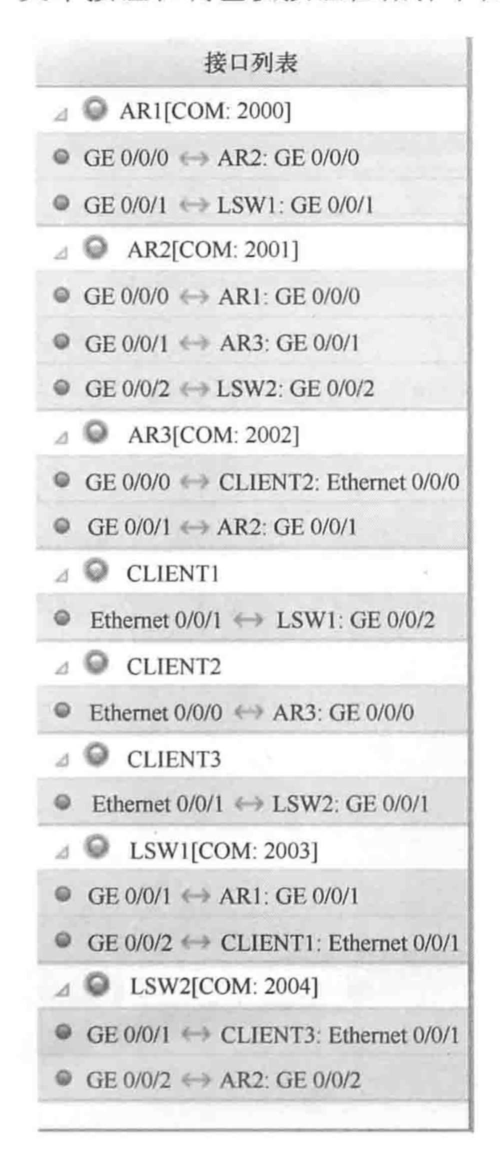


图 1-13　设备接口区

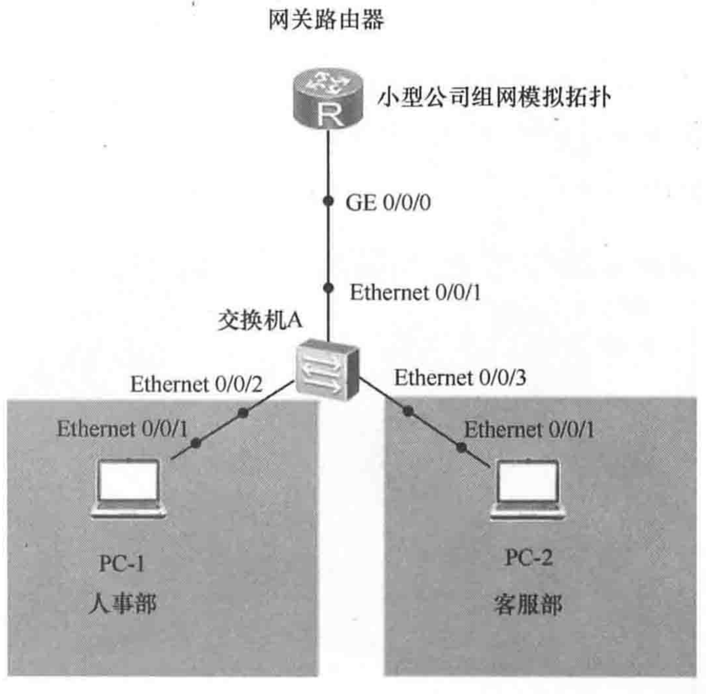


图 1-14　选择设备搭建拓扑

步骤 2：配置设备。在拓扑中的设备图标上单击鼠标右键，在弹出的快捷菜单中选择“设置”命令，打开设备接口配置界面。

在“视图”选项卡中，可以查看设备面板及可供使用的接口卡，如图 1-15 所示。如需为设备增加接口卡，可在“eNSP 支持的接口卡”区域选择合适的接口卡，直接拖至上方的设备面板上相应槽位即可；如需删除某个接口卡，直接将设备面板上的接口卡拖

回“eNSP 支持的接口卡”区域即可。注意，只有在设备电源关闭的情况下才能进行增加或删除接口卡的操作。

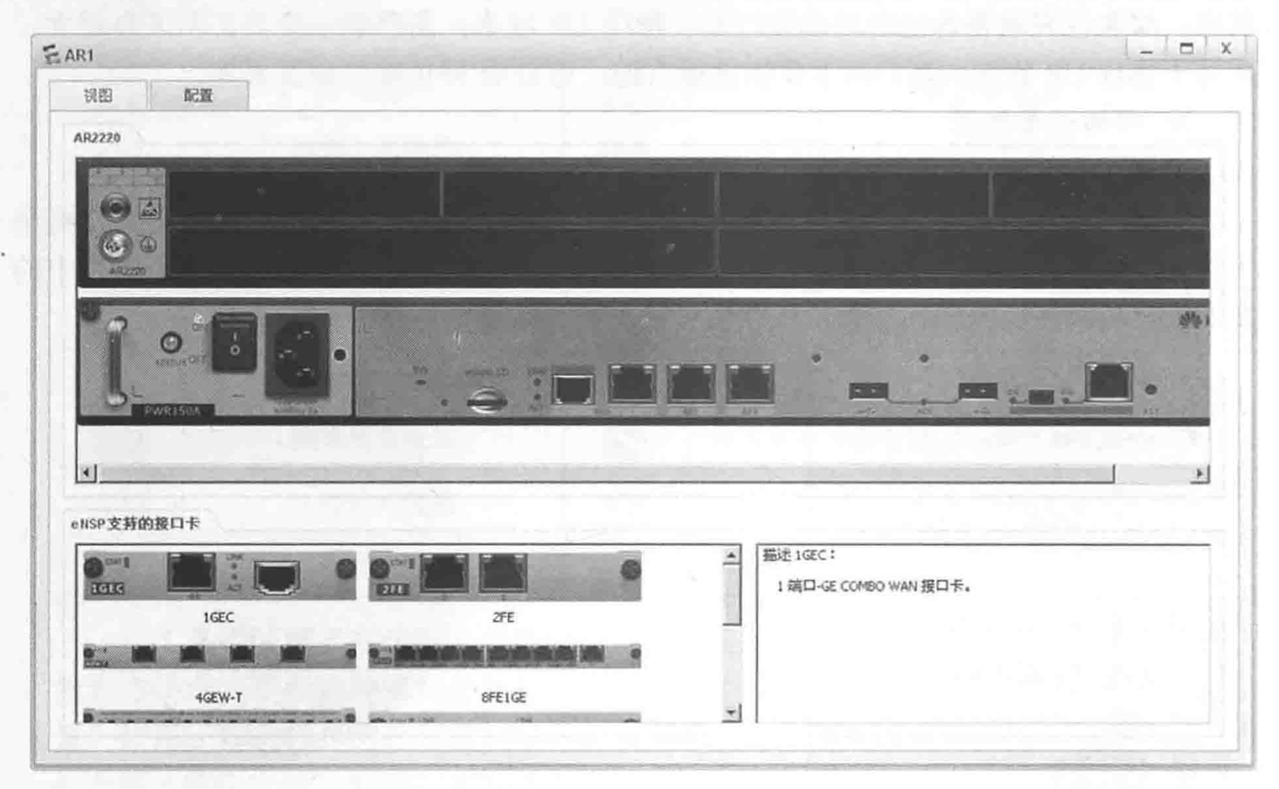


图 1-15 设备配置界面

在“配置”选项卡中，可以设置设备的串口号，串口号范围在 2000~65535，默认情况下从起始数字 2000 开始使用。可以自行更改串口号并单击“应用”按钮生效，如图 1-16 所示。

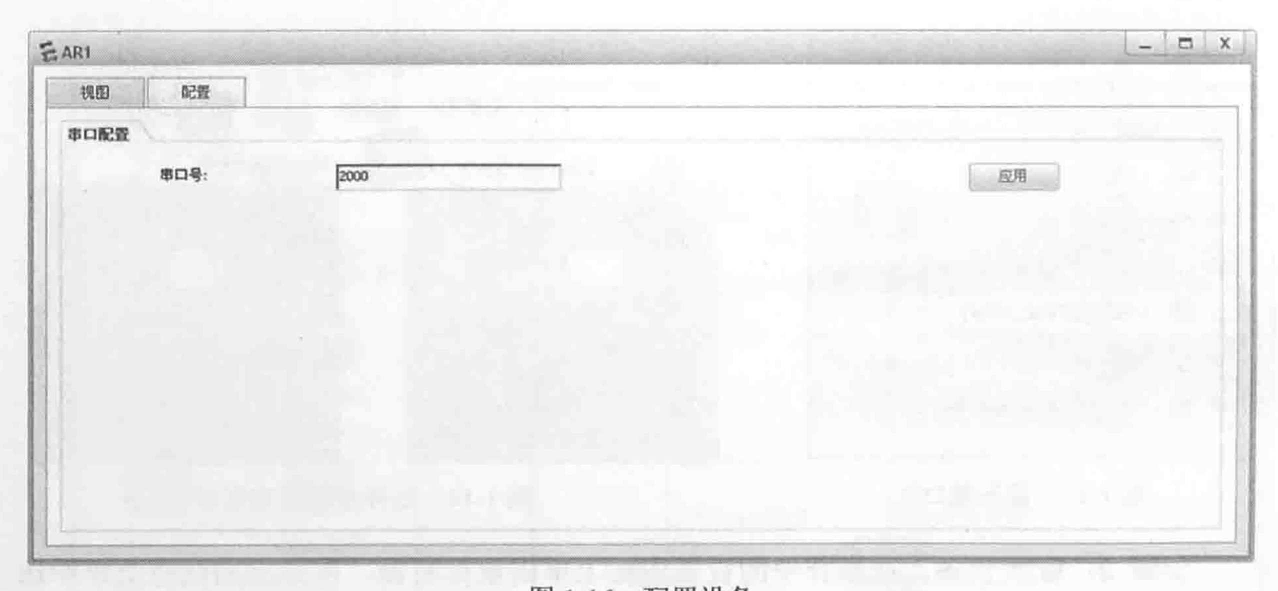


图 1-16 配置设备

在模拟 PC 上单击鼠标右键，在弹出的快捷菜单中选择“设置”命令，打开设置的对话框。在“基础配置”选项卡中配置设备的基础参数，如 IP 地址、子网掩码和 MAC 地址等，如图 1-17 所示。

CLIENT1

基础配置　命令行　组播　UDP发包工具

主机名：客户端

MAC 地址：54-89-98-CF-0A-0D

IPv4 配置

静态　DHCP　自动获取 DNS 服务器地址

IP 地址：101 . 1 . 1 . 1　DNS1: 0 . 0 . 0 . 0

子网掩码：255 . 255 . 255 . 0　DNS2: 0 . 0 . 0 . 0

网关：0 . 0 . 0 . 0

IPv6 配置

静态　DHCPv6

IPv6 地址：::

前缀长度：128

IPv6 网关：::

应用

图 1-17　PC 配置界面

在“命令行”选项卡中可以输入 **ping** 命令，测试连通性，如图 1-18 所示。

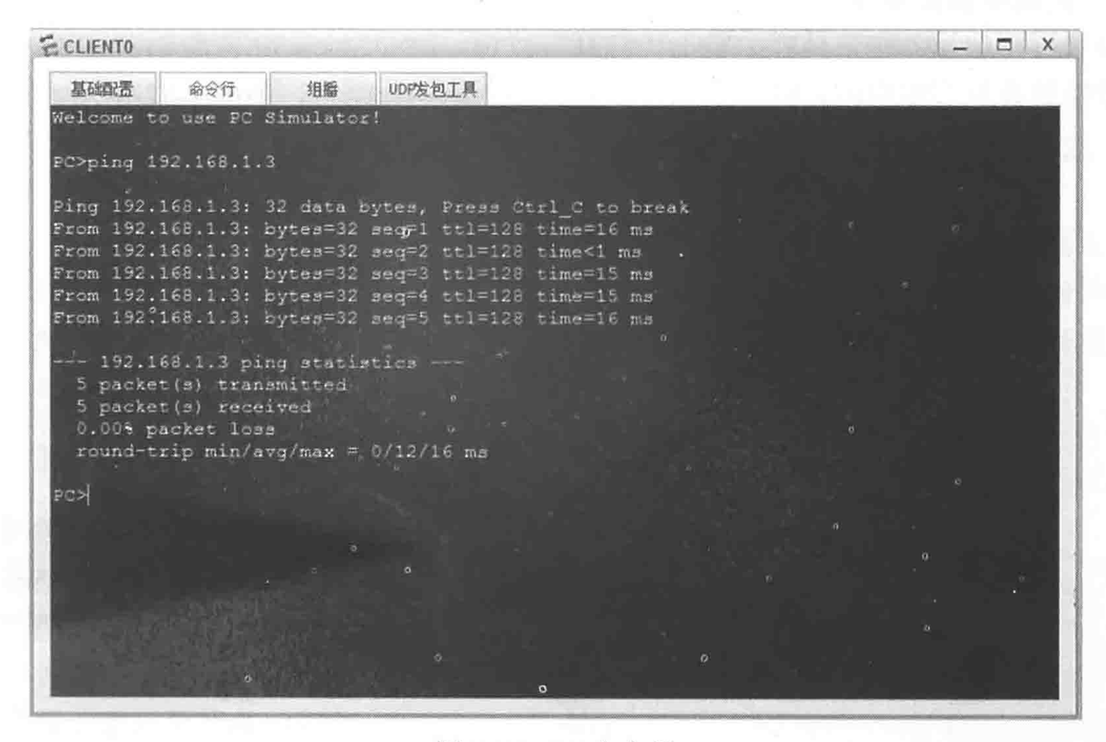


图 1-18　PC 命令行

步骤 3：设备连接。根据设备接口的不同可以灵活选择线缆的类型。当线缆仅一端连接了设备，而此时希望取消连接时，在工作区单击鼠标右键或者按<Esc>键即可。选择“Auto”可以自动识别接口卡选择相应线缆。常见的如“Copper”为双绞线，“Serial”为串口线，如图 1-19 所示。

步骤 4：配置导入。在设备未启动的状态下，在设备上单击鼠标右键，在快捷菜单中选

择“导入设备配置”命令，可以选择设备配置文件（.cfg 或者.zip 格式）并导入到设备中。

步骤 5：设备启动。选中需要启动的设备后，可以通过单击工具栏中的“启动设备”按钮或者选择该设备的右键菜单的“启动”命令来启动设备，如图 1-20 所示。启动后，双击设备图标，通过弹出的 CLI 命令行界面进行配置。

步骤 6：设备和拓扑保存。完成配置后可以单击工具栏中的“保存”按钮来保存拓扑图，并导出设备的配置文件。在设备上单击鼠标右键，在快捷菜单中选择“导出设备配置”命令，如图 1-21 所示，输入设备配置文件的文件名，并将设备配置信息导出为.cfg 文件。

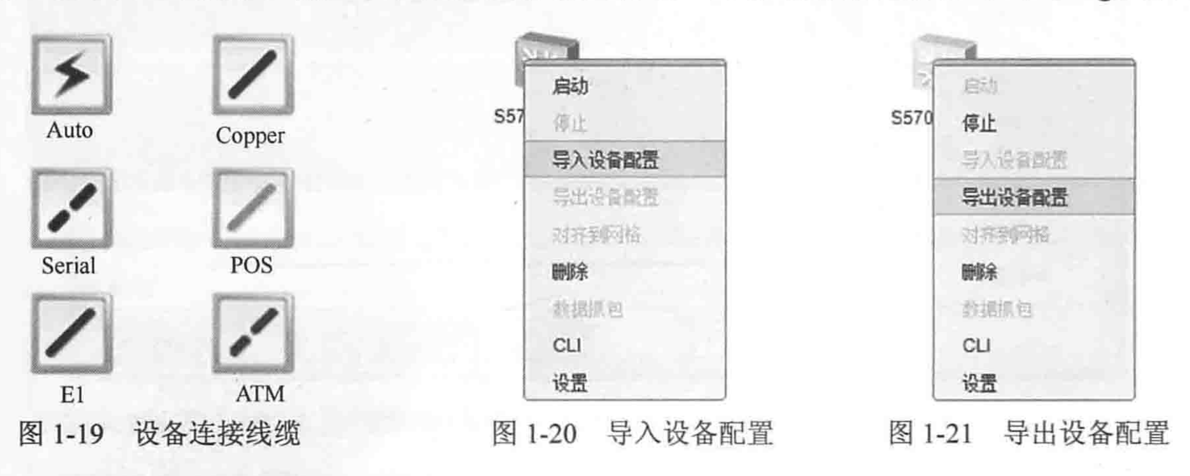


图 1-19　设备连接线缆　　图 1-20　导入设备配置　　图 1-21　导出设备配置

4. 扩展功能介绍

（1）样例加载。在工具栏中单击“打开”按钮，可弹出 eNSP 附带的验证各种网络协议特性的典型实验案例。打开样例可以看到清晰的拓扑组网，如图 1-22 所示。

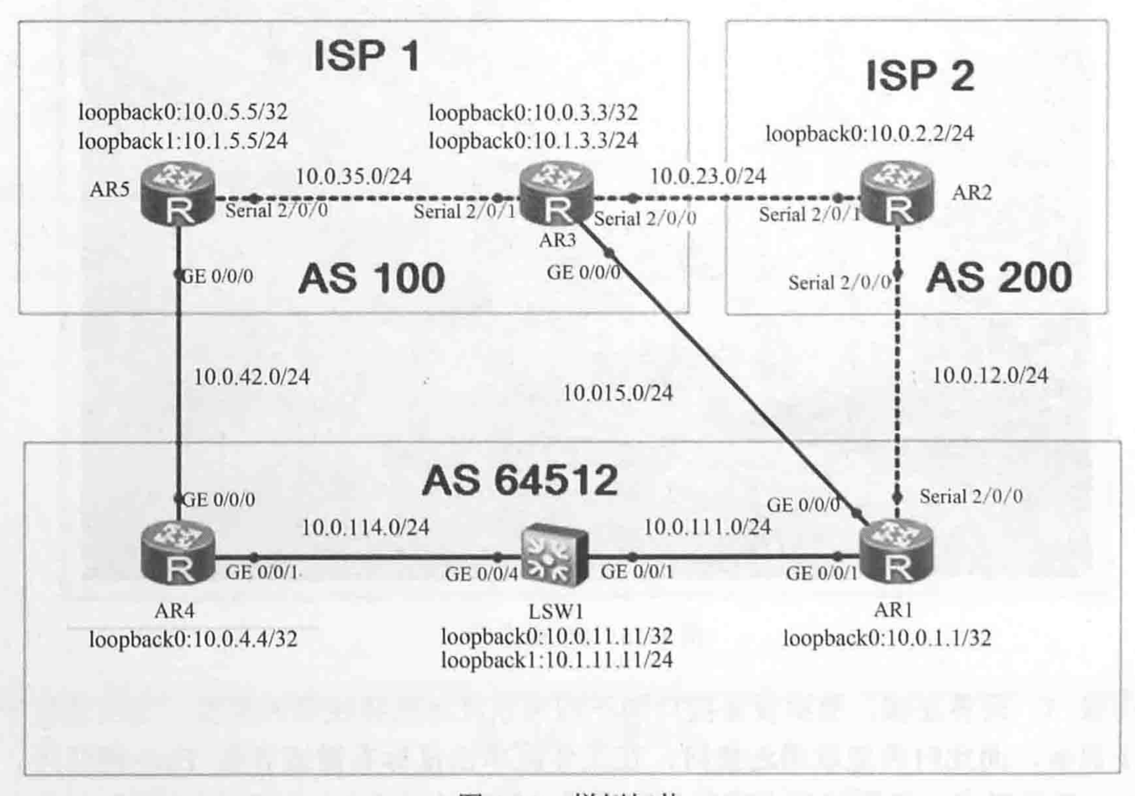


图 1-22　样例拓扑

每个样例都包括了具体的实验配置，如同一个配置好的真实的网络环境。当运行拓扑图

中所有的设备后，设备会自动加载配置，用户便可以在这套模拟环境中学习和验证理论知识。

（2）数据抓包。设备运行时，在设备接口处单击鼠标右键，在快捷菜单中可选择“数据抓包”命令。通过抓包，用户可直观感受到数据包的流动，更深刻地理解网络协议的原理。抓包的相关操作在后续实验中将详细介绍。

（3）支持与真实 PC 进行桥接。通过虚拟设备接口与真实网卡的绑定，实现虚拟设备与真实设备的对接。后续将有专门的实验介绍，或者请自行参看帮助文档。

（4）支持使用第三方软件登录 eNSP 模拟设备。在 eNSP 软件的接口视图中，设备名称后面会显示一个串口号，如图 1-23 所示。该端口号即使用第三方工具时需要设置的串口号。



图 1-23　设备串口号

1.2　熟悉 VRP 基本操作

原理概述

VRP（Versatile Routing Platform，通用路由平台）是华为公司数据通信产品的通用网络操作系统平台，拥有一致的网络界面、用户界面和管理界面。在 VRP 操作系统中，用户通过命令行对设备下发各种命令来实现对设备的配置与日常维护操作。

用户登录到路由器后出现命令行提示符后，即进入命令行接口 CLI（Command Line Interface）。命令行接口是用户与路由器进行交互的常用工具。

当用户输入命令时，如果不记得此命令的关键字或参数，可以使用命令行的帮助获取全部或部分关键字和参数的提示。用户也可以通过使用系统快捷键完成对应命令的输入，简化操作。在首次登录设备时，用户可根据需要完成设备的基本配置，如设备名称的修改、时钟的配置以及标题文本的设置等。

实验目的

- 熟悉 VRP 的基本操作
- 掌握命令行视图的切换
- 掌握命令行帮助和快捷键的使用
- 掌握修改设备名称和设置时钟的方法
- 掌握设置标题信息的方法
- 掌握查看路由器基本信息的方法

实验内容

本实验模拟用户首次使用 VRP 操作系统的过程。在登录路由器后使用命令行来配置设备，进行命令行视图的切换、命令行帮助和快捷键的使用，并完成设备的基本配置，包括修改路由器名称、配置路由器时钟、设置标题文本以及使用命令行查看路由器基本信息等。

实验拓扑

VRP 基本操作的拓扑如图 1-24 所示。



图 1-24 熟悉 VRP 基本操作拓扑

实验步骤

1. 命令视图切换

启动设备，登录设备成功后即进入用户视图，如图 1-25 所示。

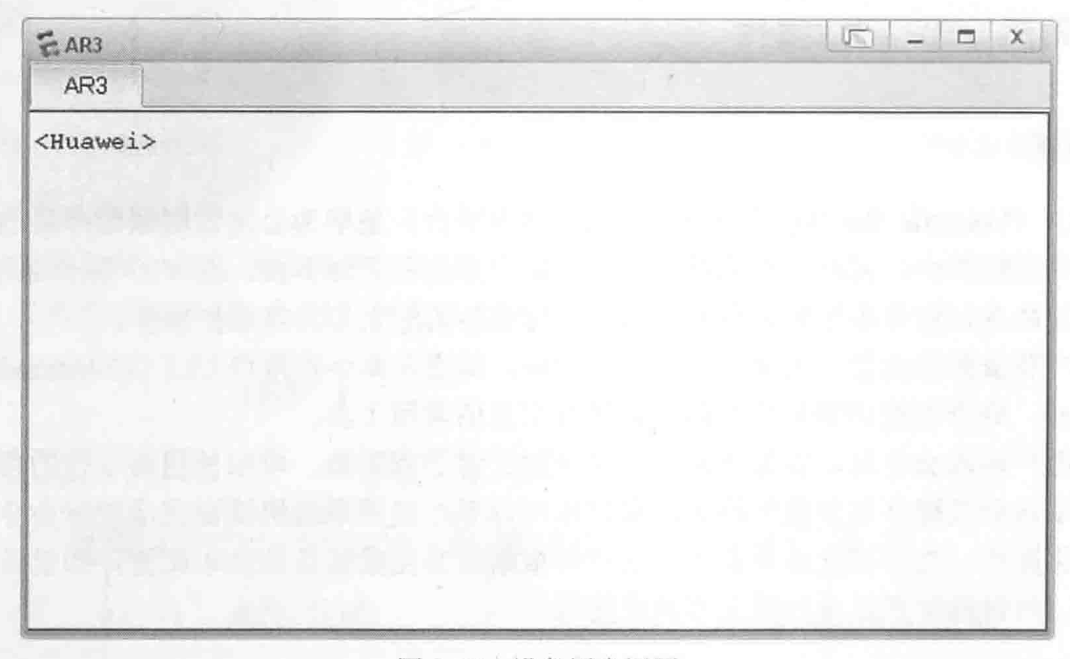


图 1-25 设备用户视图

在用户视图下只能使用参观和监控级命令，如使用 **display version** 命令显示系统软件版本及硬件等信息。

```
<Huawei>display version
Huawei Versatile Routing Platform Software
VRP (R) software, Version 5.130 (AR2200 V200R001C01SPC300)
Copyright (C) 2011 HUAWEI TECH CO., LTD
Huawei AR2220 Router uptime is 0 week, 0 day, 0 hour, 2 minutes
BKP 0 version information:
……output omit……
```

从以上内容中可以观察到 VRP 操作系统的版本、设备的型号和启动时间等信息。

在用户视图下使用 **system-view** 命令可以切换到系统视图。在系统视图下可以配置接口、协议等，使用 **quit** 命令又可以切换回用户视图。

```
<Huawei>system-view
Enter system view, return user view with Ctrl+Z.
```

```
[Huawei]quit
<Huawei>
```

在系统视图下使用相应命令可进入其他视图，如使用 **interface** 命令进入接口视图。在接口视图下可以使用 **ip address** 命令配置接口 IP 地址、子网掩码。

为路由器的 GE 0/0/0 接口配置 IP 地址时可以使用子网掩码长度，也可以使用完整的子网掩码，如掩码为 255.255.255.0，可以使用 24 替代。

```
<Huawei>system-view
Enter system view, return user view with Ctrl+Z
[Huawei]interface GigabitEthernet 0/0/0
[Huawei-GigabitEthernet0/0/0]ip address 10.1.1.1 24
```

配置完成后，可以使用 **return** 命令直接退回到用户视图。

```
[Huawei-GigabitEthernet0/0/0]return
<Huawei>
```

提示 **return** 命令可以使用户从任意非用户视图退回到用户视图，也可以用组合快捷键 **<Ctrl+Z>** 完成。

2. 命令行帮助

如果用户忘记命令的参数或关键字，可使用命令行在线帮助。命令行在线帮助分为完全帮助和部分帮助。

（1）完全帮助：在任意命令视图下，输入“？”获取该命令视图下所有的命令及其简单描述。

如在系统视图下，输入“？”获取该命令视图下所有的命令及其简单描述。

```
[Huawei]?
System view commands:
  Aaa                          <Group> aaa command group
  aaa-authen-bypass                 Set remote authentication bypass
  aaa-author-bypass                 Set remote authorization bypass
  aaa-author-cmd-bypass             Set remote command authorization bypass
  access-user                       User access
  acl                               Specify ACL configuration information
  alarm                             Alarm
  ……
```

也可以输入一个命令，后接以空格分隔的“？”，列出全部关键字或参数及其简单描述。

如在系统视图下，列出 **interface** 命令参数及其简单描述。

```
[Huawei]interface ?
  Bridge-if                Bridge-if interfac。
  Cellular                 Cellular interface
  Dialer                   Dialer interface
  Eth-Trunk                Ethernet-Trunk interface
  GigabitEthernet          GigabitEthernet interface
  ……
```

（2）部分帮助：输入一字符串，其后紧接“？”，列出以该字符串开头的所有关键字。

如在系统视图下列出以“rou”字符串开头的所有命令及其简单描述。

```
[Huawei]rou?
  Route                        Routing Module
  route-policy                 Route-policy
  route-policy-change          Specify route policy change parameter
```

```
router                          Configure router information
```

3. 快捷键使用

命令行接口提供了基本的命令编辑功能，支持多行编辑，每条命令的最大长度为 256 个字符。各功能键详细描述如下：

- 退格键<BackSpace>表示删除光标位置的前一个字符；
- 左光标键<←>或<Ctrl+B>表示光标向左移动一个字符位置；
- 右光标键<→>或<Ctrl+F>表示光标向右移动一个字符位置；
- 删除键<Delete>表示删除光标位置字符；
- 上下光标键<↑>、<↓>表示显示历史命令；
- 当用户输入不完整的关键字后按下<Tab>键，系统自动执行部分帮助，将命令补全。比如输入“dis”后，按<Tab>键可以将命令补全为“display”。

```
<R1>dis
<R1>display
```

可以通过 **display hotkey** 命令来查看已定义、未定义和系统保留的快捷键的情况。

4. 修改路由器名称

当网络上有多个设备需要管理时，用户可以为每个设备设置特定的名称，以便于管理和识别。

在系统视图下，使用 **sysname** 命令修改当前路由器名称，如更改当前路由器的系统名称为 R1。

```
[Huawei]sysname R1
[R1]
```

5. 设置路由器时钟

为了保证网络中的设备有准确的时钟信号，用户需要准确设置设备的系统时钟。

clock datetime 命令用于设置当前时间和日期；**clock timezone** 命令用于设置所在的时区。

例如：在用户视图下，使用 **clock datetime** 命令修改系统日期和时间为 2011 年 9 月 15 日 12 时。

```
<R1>clock datetime 12:00:00 2011-09-15
```

例如：在用户视图下，使用 **clock timezone** 命令，设置所在的时区为北京。

```
<R1>clock timezone BJ add 08:00:00
```

说明 系统默认是伦敦时间，而北京处于+8 时区，时间偏移量增加了 8，因此，在配置时需要加上偏移量 8，才能得到预期的北京时区。

6. 设置标题信息

如果需要对登录路由器的用户提供警示或说明信息，可以设置登录时或登录成功后的标题信息。

使用 **header login** 命令，可设置登录时的标题文本为 hello；使用 **header shell** 命令，可设置登录成功后的标题文本信息为“Welcome to Huawei certification lab”。

```
[R1]header login information "hello"
[R1]header shell information "Welcome to Huawei certification lab"
```

其中，login 参数为用户在登录路由器认证过程中激活终端连接时显示的标题信息，是用户在连接到路由器并进行登录验证以及开始配置时，系统所显示的一段提示信息，当需要为用户登录提供明确的提示信息时，可以使用此配置。Shell 参数为当用户成功登

录到路由器上，已经建立了会话时显示的标题信息。

配置完成后，尝试退出路由器命令行界面重新登录，即可观察到欢迎信息。

```
[R1]quit
<R1>quit

Configuration console exit, please retry to log on
Password:
Welcome to Huawei certification lab
<R1>
```



通常情况下，登录标语信息用于警告非法登录或者说明信息。

7. 查看路由器基本信息

使用 display 系列命令可查看路由器基本信息或运行状态。

使用 **display version** 命令查看路由器信息。

```
<R1>display version
Huawei Versatile Routing Platform Software
VRP (R) software, Version 5.130 (AR2200 V200R003C00)
Copyright (C) 2011-2012 HUAWEI TECH CO., LTD
Huawei AR2220 Router uptime is 0 week, 0 day, 0 hour, 17 minutes
BKP 0 version information:
……
```

可以观察到 VRP 操作系统的版本、设备的型号、启动时间等信息。

使用 **display current-configuration** 命令查看路由器当前配置。

```
<R1>display current-configuration
[V200R003C00]
#
 sysname R1
 header shell information "Welcome to Huawei certification lab"
 header login information "hello"
#
 snmp-agent local-engineid 800007DB03000000000000
 snmp-agent
#
 clock timezone BJ add 08:00:00
 clock daylight-saving-time Day Light Saving Time repeating 12:32 9-1 12:32 11-23 00:00 2005 2005
#
……
#
Return
```

可以观察到所有路由器的已配置信息。

使用 **display interface GigabitEthernet 0/0/0** 命令查看路由器 GE 0/0/0 接口的状态信息。

```
[R1]display interface GigabitEthernet 0/0/0
GigabitEthernet0/0/0 current state : DOWN
Line protocol current state : DOWN
Description:HUAWEI, AR Series, GigabitEthernet0/0/0 Interface
Route Port,The Maximum Transmit Unit is 1500
Internet Address is 10.1.1.1/24
IP Sending Frames' Format is PKTFMT_ETHNT_2, Hardware address is 00e0-fc03-3a27
……
```

可以观察到该接口的物理状态、接口 IP 地址以及其他的统计信息。

1.3 熟悉常用的 IP 相关命令

原理概述

华为设备支持多种配置方式，包括 Web 界面管理等。但作为一名网络工程师，必须熟悉使用命令行的方式进行设备管理。在工作中，对路由器和交换机最常用的操作命令就是 IP 相关命令，如配置主机名、IP 地址、测试 IP 数据包连通性等。这些命令是基本的配置和测试命令。

实验目的

- 掌握路由器命名的方法
- 掌握配置路由器 IP 地址方法
- 掌握测试 IP 地址连通性的方法
- 掌握查看设备配置的方法
- 掌握抓包的方法

实验内容

本实验模拟简单的企业网络场景，某公司购买了新的路由器和交换机。交换机 S1 连接客服部 PC-1，S2 连接市场部 PC-2，路由器 R1 连接 S1 和 S2 两台交换机。网络管理员需要首先熟悉设备的使用，包括基础的 IP 配置和查看命令。

实验拓扑

常用的 IP 相关命令的拓扑如图 1-26 所示。

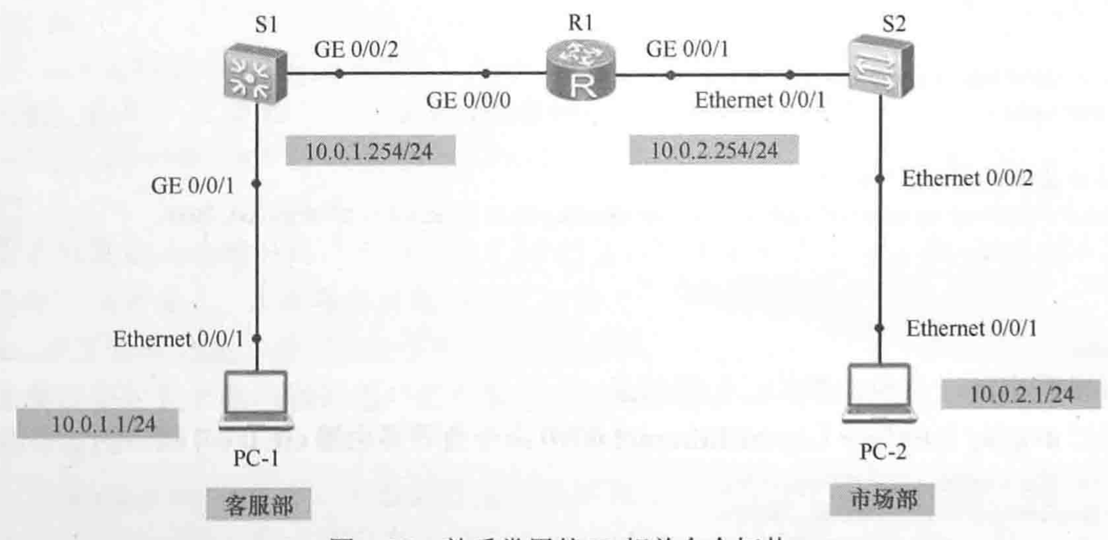


图 1-26 熟悉常用的 IP 相关命令拓扑

实验编址

实验编址见表 1-2。

表 1-2　实验编址

设备	接口	IP 地址	子网掩码	默认网关
PC-1	Ethernet 0/0/1	10.0.1.1	255.255.255.0	10.0.1.254
PC-2	Ethernet 0/0/1	10.0.2.1	255.255.255.0	10.0.2.254
R1（AR2220）	GE 0/0/0	10.0.1.254	255.255.255.0	N/A
	GE 0/0/1	10.0.2.254	255.255.255.0	N/A

实验步骤

1. 基本配置

根据实验编址，使用图形化界面配置 PC 的 IP 地址，以客服部 PC-1 为例，如图 1-27 所示。



图 1-27　PC-1 配置界面

PC-1 设置完成后，继续设置市场部 PC-2，如图 1-28 所示。

图 1-28　PC-2 配置界面

配置路由器的主机名，打开 R1 的命令行界面，即进入用户视图。在用户视图下，用户可以完成查看运行状态和统计信息等功能。此时屏幕上显示：

```
<Huawei>
```

路由器主机名为默认的主机名 Huawei。要更改主机名，必须使用进入系统视图模式。在系统视图下，**system-view** 命令用户可以配置系统参数以及通过该视图进入其他的功能配置视图。

```
<Huawei>system-view
[Huawei]
```

这时图标由<Huawei>变成了[Huawei]，表示进入了系统视图模式。

在系统视图下，使用 **sysname** 命令修改设备主机名为 R1。

```
[Huawei]sysname R1
```

可以观察到，主机名由原来的[Huawei]变成了[R1]，表示主机名修改成功。使用 **quit** 命令退出当前模式。

```
[R1]quit
```

此时[R1]标志已经变成<R1>，表明已经成功退回到用户视图。在用户视图下使用 **save** 命令保存当前配置。

```
<R1>save
```

这时会提示是否继续保存，输入 **"y"** 确认保存动作。

```
<R1>save
  The current configuration will be written to the device.
Are you sure to continue? (y/n)[n]:y
  It will take several minutes to save configuration file, please wait........
Configuration file had been saved successfully
  Note: The configuration file will take effect after being activated
```

出现以上信息表示保存成功。

2. 配置路由器接口 IP 地址

从系统视图进入接口视图，在该视图下配置接口相关的物理属性、链路层特性及 IP 地址等重要参数。使用 **interface** 命令进入路由器相应的接口视图 GE 0/0/0。

```
[R1]interface GigabitEthernet 0/0/0
```

在路由器的接口视图下配置路由器接口 IP 地址和掩码。注意，华为设备上的物理接口默认都处于开启状态。

```
[R1-GigabitEthernet0/0/0]ip address 10.0.1.254 255.255.255.0
```

配置完成后，使用 **display ip interface brief** 命令查看接口与 IP 相关摘要信息。

```
[R1-GigabitEthernet0/0/0]display ip interface brief
*down: administratively down
^down: standby
(l): loopback
(s): spoofing
The number of interface that is UP in Physical is 3
The number of interface that is DOWN in Physical is 1
The number of interface that is UP in Protocol is 2
The number of interface that is DOWN in Protocol is 2
Interface                    IP Address/Mask     Physical    Protocol
GigabitEthernet0/0/0          10.0.1.254/24        up          up
GigabitEthernet0/0/1          unassigned           up          down
GigabitEthernet0/0/2          unassigned           down        down
NULL0                         unassigned           up          up(s)
```

可以观察到，路由器接口 GE 0/0/0 的 IP 地址已经配置完成，"Physical" 为 UP，即接口的物理状态处于正常启动的状态；"Protocol" 为 UP，即接口的链路协议状态处于正

常启动的状态。

同理配置路由器 GE 0/0/1 的 IP 地址。如果在配置过程中对命令非常熟悉，可以采用简写的方式配置。

```
<R1>system-view
[R1]int g0/0/1
[R1-GigabitEthernet0/0/1]ip add 10.0.2.254 24
```

注意，即便是简写，也要保证所输入的命令关键字是唯一的，否则不会成功。

如果忘记命令，可以输入“？”查看相关命令。如果输入命令首部分，可以使用<Tab>键选择性补齐命令。

```
[Huawei]inter?
  interface    Specify the interface configuration view
[Huawei]inter
[Huawei]interface
```

配置完成后，再次确认接口与 IP 相关摘要信息。

```
<R1>display ip interface brief
*down: administratively down
^down: standby
(l): loopback
(s): spoofing
The number of interface that is UP in Physical is 3
The number of interface that is DOWN in Physical is 1
The number of interface that is UP in Protocol is 3
The number of interface that is DOWN in Protocol is 1
Interface                    IP Address/Mask      Physical     Protocol
GigabitEthernet0/0/0          10.0.1.254/24        up           up
GigabitEthernet0/0/1          10.0.2.254/24        up           up
GigabitEthernet0/0/2          unassigned           down         down
NULL0                         unassigned           up           up(s)
```

可以观察到，路由器 GE 0/0/0 与 GE 0/0/1 的接口 IP 地址已经配置完成。物理接口工作正常，接口的链路协议状态处于正常启动的状态。

3. 查看路由器配置信息

经过以上步骤的配置，路由器接口的 IP 地址已经配置完成，可以使用 **display ip routing-table** 命令查看 IPv4 路由表的信息。

```
<R1>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
Destinations : 10 Routes : 10
Destination/MaskProtoPreCostFlagsNextHopInterface
10.0.1.0/24          Direct  0    0      D   10.0.1.254   GigabitEthernet0/0/0
10.0.1.254/32        Direct  0    0      D   127.0.0.1    GigabitEthernet0/0/0
10.0.1.255/32        Direct  0    0      D   127.0.0.1    GigabitEthernet0/0/0
10.0.2.0/24          Direct  0    0      D   10.0.2.254   GigabitEthernet0/0/1
10.0.2.254/32        Direct  0    0      D   127.0.0.1    GigabitEthernet0/0/1
10.0.2.255/32        Direct  0    0      D   127.0.0.1    GigabitEthernet0/0/1
127.0.0.0/8          Direct  0    0      D   127.0.0.1    InLoopBack0
127.0.0.1/32         Direct  0    0      D   127.0.0.1    InLoopBack0
127.255.255.255/32   Direct  0    0      D   127.0.0.1    InLoopBack0
255.255.255.255/32   Direct  0    0      D   127.0.0.1    InLoopBack0
```

可以观察到，路由器 R1 在 GE 0/0/0 接口上直连了一个 10.0.1.0/24 的网段，在 GE 0/0/1 接口上直连了一个 10.0.2.0/24 的网段。

其中，“Route Flags”为路由标记，“R”表示该路由是迭代路由，“D”表示该路由下发到 FIB 表。“Routing Tables：Public”表示此路由表是公网路由表，如果是私网路由表，则显示私网的名称，如 Routing Tables: ABC。“Destinations”表示目的网络/主机的总数。“Routes”表示路由的总数。“Destination/Mask”表示目的网络/主机的地址和掩码长度，“Proto”表示接收此路由的路由协议，“Direct”表示直连路由，“Pre”表示此路由的优先级，“Cost”表示此路由的路由开销值。“NextHop”表示此路由的下一跳地址，“Interface”表示此路由下一跳的出接口。使用 **Ping** 命令测试路由器 R1 与 PC 间的连通性。下面以测试去往 PC-1 的连通性为例说明。

```
<R1>ping 10.0.1.1
  PING 10.0.1.1: 56    data bytes, press CTRL_C to break
    Reply from 10.0.1.1: bytes=56 Sequence=1 ttl=128 time=200 ms
    Reply from 10.0.1.1: bytes=56 Sequence=2 ttl=128 time=70 ms
    Reply from 10.0.1.1: bytes=56 Sequence=3 ttl=128 time=40 ms
    Reply from 10.0.1.1: bytes=56 Sequence=4 ttl=128 time=1 ms
    Reply from 10.0.1.1: bytes=56 Sequence=5 ttl=128 time=10 ms
  --- 10.0.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 1/64/200 ms
```

若显示上述结果，则表明测试连通性正常。

```
<R1>ping 10.0.1.1
  PING 10.0.1.1: 56    data bytes, press CTRL_C to break
    Request time out
    Request time out
    Request time out
    Request time out
    Request time out
  --- 10.0.1.1 ping statistics ---
    5 packet(s) transmitted
    0 packet(s) received
100.00% packet loss
```

若显示上述结果，则表示连通性测试失败，即 R1 与 PC-1 连通性异常。

直连网段连通性测试完毕后，测试非直连设备的连通性，即 PC-1 与 PC-2 的连通性。双击设备打开配置界面，单击“命令行”选项卡。此命令行如同 PC 的 DOS 窗口一样，可执行基本命令，如图 1-29 所示。

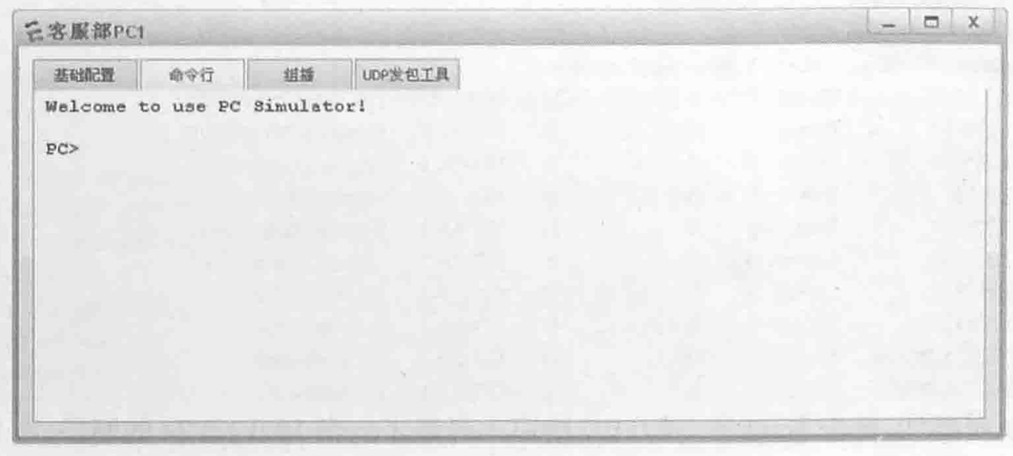


图 1-29 PC-1 配置界面

测试 PC-1 到 PC-2 间的连通性。

```
PC>ping 10.0.2.1
Ping 10.0.2.1: 32 data bytes, Press Ctrl_C to break
From 10.0.2.1: bytes=32 seq=1 ttl=127 time=47 ms
From 10.0.2.1: bytes=32 seq=2 ttl=127 time=47 ms
From 10.0.2.1: bytes=32 seq=3 ttl=127 time=47 ms
From 10.0.2.1: bytes=32 seq=4 ttl=127 time=62 ms
From 10.0.2.1: bytes=32 seq=5 ttl=127 time=32 ms
--- 10.0.2.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 32/47/62 ms
```

可以观察到 PC-1 与 PC-2 之间能正常通信。

4. 使用抓包工具

以抓取 R1 上 GE 0/0/0 接口的数据包为例，在 R1 与 S1 的直连链路上，在接口 GE 0/0/0 上单击鼠标右键，在弹出的快捷菜单中选择“开始抓包”命令，如图 1-30 所示。



图 1-30　通过右键菜单选择抓包

这时会显示出解包的结果，如图 1-31 所示。

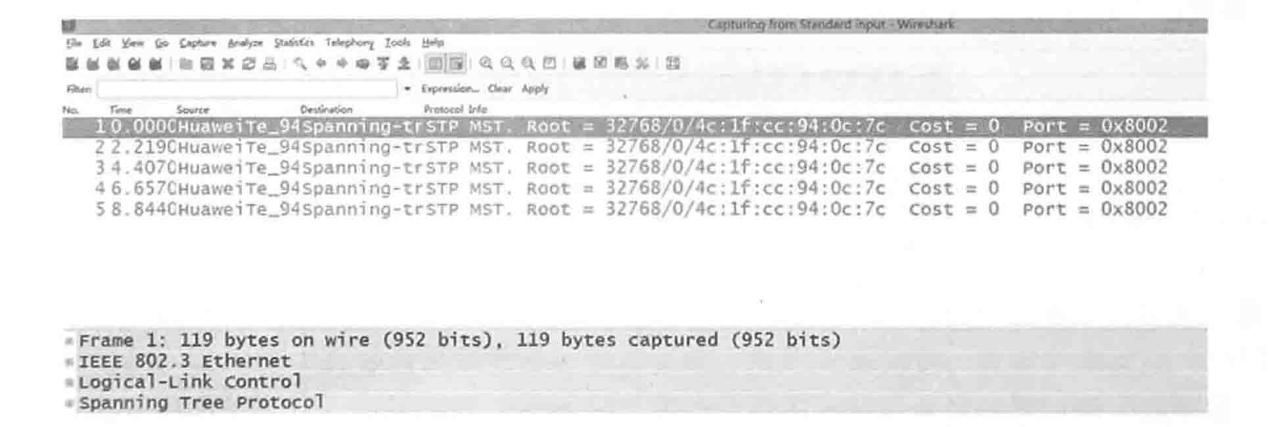


图 1-31　解包结果

双击数据包可查看详细的数据包内容，如图 1-32 所示。

如果不需要继续抓包，可在接口的快捷菜单中选择“停止抓包”命令。

用户还可以采取另一种方式来抓包，效果相同。

单击界面上方工具栏中的“数据抓包”按钮，这时会出现当前可抓取的接口列表，如图 1-33 所示。

```
Frame 39: 119 bytes on wire (952 bits), 119 bytes captured (952 bits)
IEEE 802.3 Ethernet
 Destination: Spanning-tree-(for-bridges)_00 (01:80:c2:00:00:00)
 Source: HuaweiTe_94:0c:7c (4c:1f:cc:94:0c:7c)
 Length: 105
Logical-Link Control
 DSAP: Spanning Tree BPDU (0x42)
 IG Bit: Individual
 SSAP: Spanning Tree BPDU (0x42)
 CR Bit: Command
 Control field: U, func=UI (0x03)
Spanning Tree Protocol
 Protocol Identifier: Spanning Tree Protocol (0x0000)
 Protocol Version Identifier: Multiple Spanning Tree (3)
 BPDU Type: Rapid/Multiple Spanning Tree (0x02)
 BPDU flags: 0x7c (Agreement, Forwarding, Learning, Port Role: Designated)
 Root Identifier: 32768 / 0 / 4c:1f:cc:94:0c:7c
 Root Path Cost: 0
 Bridge Identifier: 32768 / 0 / 4c:1f:cc:94:0c:7c
 Port identifier: 0x8002
 Message Age: 0
 Max Age: 20
 Hello Time: 2
 Forward Delay: 15
 Version 1 Length: 0
 Version 3 Length: 64
 MST Extension
```

图 1-32　数据包详细内容



图 1-33　工具栏选择抓包

以抓取 R1 上 GE 0/0/0 接口的数据包为例，在“选择设备”栏中选择“R1”，在“选择接口”栏中选择“GE 0/0/0”，然后单击“开始抓包”按钮，如图 1-34 所示。



图 1-34　抓包设置界面

这时会显示出如图 1-31 所示的解包结果。同样双击数据包查看详细的数据包内容。如图 1-35 所示。

```
Frame 39: 119 bytes on wire (952 bits), 119 bytes captured (952 bits)
IEEE 802.3 Ethernet
  Destination: Spanning-tree-(for-bridges)_00 (01:80:c2:00:00:00)
  Source: HuaweiTe_94:0c:7c (4c:1f:cc:94:0c:7c)
  Length: 105
Logical-Link Control
  DSAP: Spanning Tree BPDU (0x42)
  IG Bit: Individual
  SSAP: Spanning Tree BPDU (0x42)
  CR Bit: Command
  Control field: U, func=UI (0x03)
Spanning Tree Protocol
  Protocol Identifier: Spanning Tree Protocol (0x0000)
  Protocol Version Identifier: Multiple Spanning Tree (3)
  BPDU Type: Rapid/Multiple Spanning Tree (0x02)
  BPDU flags: 0x7c (Agreement, Forwarding, Learning, Port Role: Designated)
  Root Identifier: 32768 / 0 / 4c:1f:cc:94:0c:7c
  Root Path Cost: 0
  Bridge Identifier: 32768 / 0 / 4c:1f:cc:94:0c:7c
  Port identifier: 0x8002
  Message Age: 0
  Max Age: 20
  Hello Time: 2
  Forward Delay: 15
  Version 1 Length: 0
  Version 3 Length: 64
  MST Extension
```

图 1-35　数据包详细内容

如果不需要继续抓包，单击“停止抓包”按钮即可。

思考

管理员要经常在路由器上使用 **display ip interface brief** 命令查看接口状态，但该命令若完整输入则较长，思考如何使用最简化且准确的方式输入这条命令？

1.4　配置通过 Telnet 登录系统

原理概述

Telnet（Telecommunication Network Protocol）起源于 ARPANET，是最早的 Internet 应用之一。

Telnet 通常用在远程登录应用中，以便对本地或远端运行的网络设备进行配置、监控和维护。如网络中有多台设备需要配置和管理，用户无需为每一台设备都连接一个用户终端进行本地配置，可以通过 Telnet 方式在一台设备上对多台设备进行管理或配置。如果网络中需要管理或配置的设备不在本地时，也可以通过 Telnet 方式实现对网络中设备的远程维护，极大地提高了用户操作的灵活性。

实验目的

- 理解 Telnet 的应用场景
- 掌握 Telnet 的基本配置
- 掌握 Telnet 密码验证的配置
- 掌握 Telnet 用户级别的修改方法

实验内容

本实验模拟公司网络场景。路由器 R1 是公司机房的一台设备，公司员工的办公区与机房不在同一个楼层，路由器 R2 和 R3 模拟员工主机，通过交换机 S1 与机房设备相连。为了方便用户的管理，现需要在路由器 R1 上配置 Telnet 使用户能在办公区远程管理机房设备。为了提高安全性，Telnet 需要使用密码认证，只有网络管理员能对设备进行配置和管理，普通用户仅能监控设备。

实验拓扑

配置通过 Telnet 登录系统的拓扑如图 1-36 所示。

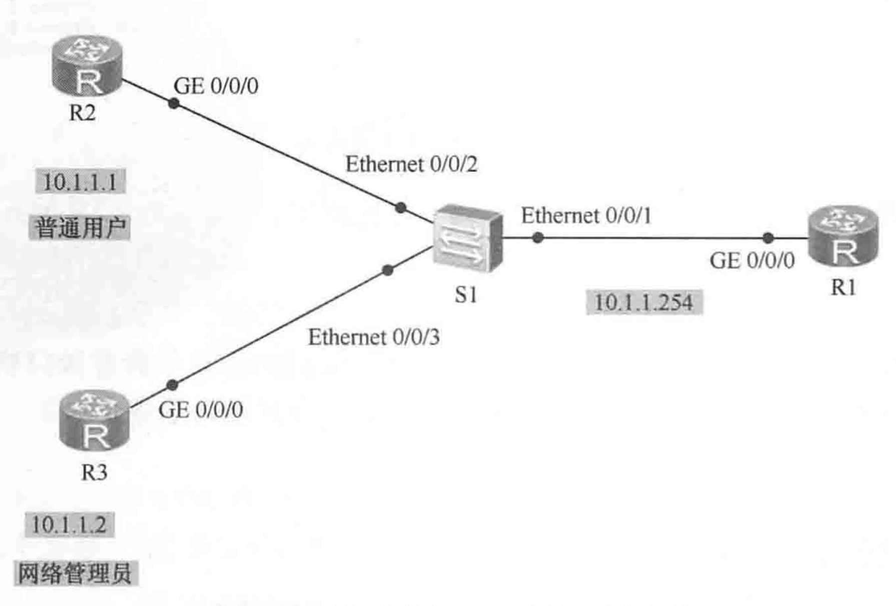


图 1-36 配置通过 Telnet 登录系统拓扑

实验编址

实验编址见表 1-3。

表 1-3 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR2220）	GE 0/0/0	10.1.1.254	255.255.255.0	N/A
R2（AR2220）	GE 0/0/0	10.1.1.1	255.255.255.0	10.1.1.254
R3（AR2220）	GE 0/0/0	10.1.1.2	255.255.255.0	10.1.1.254

实验步骤

1. 基本配置

根据实验编址进行相应的基本配置，并使用 **ping** 命令检测各直连链路的连通性。

这里以用户主机和默认网关间的连通性为例。

```
R2>ping 10.1.1.254
Ping 10.1.1.254: 32 data bytes, Press Ctrl_C to break
```

```
From 10.1.1.254: bytes=32 seq=1 ttl=255 time=47 ms
From 10.1.1.254: bytes=32 seq=2 ttl=255 time=32 ms
From 10.1.1.254: bytes=32 seq=3 ttl=255 time=47 ms
From 10.1.1.254: bytes=32 seq=4 ttl=255 time=31 ms
From 10.1.1.254: bytes=32 seq=5 ttl=255 time=31 ms
--- 10.1.1.254 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 30/68/120 ms
```

2. 配置 Telnet 的密码验证

为了方便公司员工对机房设备进行远程管理和维护，首先需要在路由器上配置 Telnet 功能。为了提高网络安全性，可在使用 Telnet 时进行密码认证，只有通过认证的用户才有权限登录设备。

在 R1 上配置 Telnet 验证方式为密码验证方式，密码为 huawei，并设置验证密码以密文方式存储，在配置文件中以加密的方式显示密码，能够使密码不容易被泄露。

```
[R1]user-interface vty 0 4
[R1-ui-vty0-4]authentication-mode password
Please configure the login password (maximum length 16):huawei
```

在用户设备 R2 和 R3 上使用 Telnet 连接 R1。

```
<R2>telnet 10.1.1.254
Trying 10.1.1.254 ...
Press CTRL+K to abort
Connected to 10.1.1.254 ...
Login authentication
Password:
Info: The max number of VTY users is 10, and the number
      of current VTY users on line is 1.
      The current login time is 2013-06-25 13:56:34.
<R1>

<R3>telnet 10.1.1.254
Trying 10.1.1.254 ...
Press CTRL+K to abort
Connected to 10.1.1.254 ...
Login authentication
Password:
Info: The max number of VTY users is 10, and the number
      of current VTY users on line is 1.
      The current login time is 2013-06-25 19:08:01.
<R1>
```

可以观察到 R2 和 R3 在连接 R1 的过程中，要求输入认证密码，只有当输入正确的密码后才能进入 R1 的用户界面。

登录成功后，可以继续使用 **display users** 命令查看已经登录的用户信息。

```
[R1]display users
  User-Intf    Delay     Type   Network Address     AuthenStatus    AuthorcmdFlag
+ 0    CON 0    00:00:00          no        Username : Unspecified
 34   VTY 0    00:01:26   TEL     10.1.1.2                  pass            no          U
sername : Unspecified
 35   VTY 1    00:00:30   TEL     10.1.1.1                  pass            no          U
sername : Unspecified
```

3. 配置 Telnet 区分不同用户的权限

为了进一步保证网络的安全性及稳定性，避免员工错误更改设备的配置，公司要求普通员工只能拥有设备的监控权限，只有网络管理员拥有设备的配置和管理权限。默认情况下，VTY 用户界面的用户级别为 0（参观级），只能使用 **ping**、**tracert** 等网络诊断命令。

在 R1 上配置 Telnet 的用户级别为 1（监控级）。普通员工仅使用密码登录设备，只能使用 **display** 等命令监控设备。

```
[R1]user-interface vty 0 4
[R1-ui-vty0-4]authentication-mode password
[R1-ui-vty0-4]set authentication password cipher huawei
[R1-ui-vty0-4]user privilege level 1
```

配置完成后，将 R2 模拟成普通用户设备，测试到 R1 的 Telnet 连接。

```
<R2>telnet 10.1.1.254
Trying 10.1.1.254 ...
Press CTRL+K to abort
Connected to 10.1.1.254 ...
Login authentication
Password:
<R1>
<R1>system-view
    ^
Error: Unrecognized command found at '^' position.
```

可以观察到，此时输入正确的密码后即可进入 R1 的用户视图，但是在试图进入 R1 的系统视图时被拒绝了，这是因为用户级别不够，所以无法执行更高一级的命令。

管理员使用自己单独的用户名和密码登录设备，拥有设备的配置和管理权限。这里要将 VTY 用户界面的认证模式修改成 AAA 认证，这样才能使用本地的用户名和密码进行认证。默认情况下，设备的 AAA 认证功能是开启的，所以只需要为管理员在本地配置相应的用户名和密码即可。

下面模拟进入 AAA 视图下配置本地用户名 admin 和密文密码 hello，并且将该用户的用户级别修改为 3（管理级）。

```
[R1]aaa
[R1-aaa]local-user admin password cipher hello privilege level 3
```

配置该用户的接入类型为 Telnet。

```
[R1-aaa]local-user admin service-type telnet
```

接下来进入 VTY 用户界面视图下，将认证模式改成 AAA。

```
[R1]user-interface vty 0 4
[R1-ui-vty0-4]authentication-mode aaa
```

将 R3 模拟成管理员用户设备，测试到 R1 的 Telnet 连接。

```
<R3>telnet 10.1.1.254
Trying 10.1.1.254 ...
Press CTRL+K to abort
Connected to 10.1.1.254 ...
Login authentication
Username:admin
Password:
Info: The max number of VTY users is 10, and the number
        of current VTY users on line is 1.
        The current login time is 2013-06-24 12:28:38.
```

```
<R1>
<R1>system-view
Enter system view, return user view with Ctrl+Z.
[R1]
```

可以观察到，此时在连接 R1 时需要同时输入用户名和密码进行认证。输入正确的用户名和密码后即可进入 R1 的用户视图下，且可以使用 **system-view** 命令进入到 R1 的系统视图下，从而对 R1 进行所有相关的配置和管理操作。

思考

Telnet 是基于 TCP 协议还是 UDP 协议的应用？为什么？

Telnet 应用安全吗？为什么？

1.5　配置通过 STelnet 登录系统

原理概述

由于 Telnet 缺少安全的认证方式，而且传输过程采用 TCP 进行明文传输，存在很大的安全隐患，单纯提供 Telnet 服务容易招致主机 IP 地址欺骗、路由欺骗等恶意攻击。传统的 Telnet 和 FTP 等通过明文传送密码和数据的方式，已经慢慢不被接受。

STelnet 是 Secure Telnet 的简称。在一个传统不安全的网络环境中，服务器通过对用户端的认证及双向的数据加密，为网络终端访问提供安全的 Telnet 服务。

SSH（Secure Shell）是一个网络安全协议，通过对网络数据的加密，使其能够在一个不安全的网络环境中，提供安全的远程登录和其他安全网络服务。SSH 特性可以提供安全的信息保障和强大的认证功能，以保护路由器不受诸如 IP 地址欺诈、明文密码截取等攻击。SSH 数据加密传输，认证机制更加安全，而且可以代替 Telnet，已经被广泛使用，成为了当前重要的网络协议之一。

SSH 基于 TCP 协议 22 端口传输数据，支持 Password 认证。用户端向服务器发出 Password 认证请求，将用户名和密码加密后发送给服务器；服务器将该信息解密后得到用户名和密码的明文，与设备上保存的用户名和密码进行比较，并返回认证成功或失败的消息。

SFTP 是 SSH File Transfer Protocol 的简称，在一个传统不安全的网络环境中，服务器通过对用户端的认证及双向的数据加密，为网络文件传输提供了安全的服务。

实验目的

- 理解 SSH 的应用场景
- 理解 SSH 协议的原理
- 掌握配置 SSH Password 认证的方法
- 掌握 SFTP 的配置

实验内容

使用路由器 R1 模拟 PC，作为 SSH 的 Client；路由器 R2 作为 SSH 的 Server，模拟远程用户端 R1 通过 SSH 协议远程登录到路由器 R2 上进行各种配置。本实验将通过 Password 认证方式来实现。

实验拓扑

配置通过 STelnet 登录系统的拓扑如图 1-37 所示。

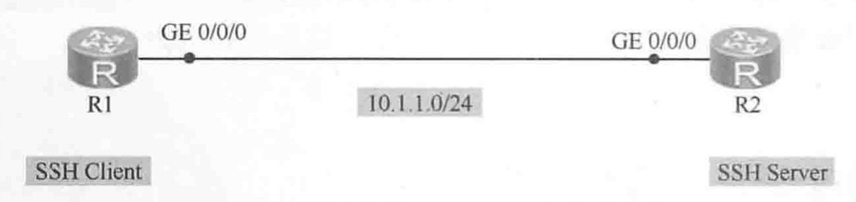


图 1-37 配置通过 STelnet 登录系统拓扑

实验编址

实验编址见表 1-4。

表 1-4 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR2220）	GE 0/0/0	10.1.1.1	255.255.255.0	N/A
R2（AR2220）	GE 0/0/0	10.1.1.2	255.255.255.0	N/A

实验步骤

1. 基本配置

由于 eNSP 模拟软件自带 PC 没有 SSH 用户端，本实验采用两台路由器模拟实验，路由器 R1 作为 SSH 的 Client，路由器 R2 作为 SSH 的 Server。

根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测各直连链路的连通性。

```
[R2]ping 10.1.1.1
  PING 10.1.1.1: 56   data bytes, press CTRL_C to break
    Reply from 10.1.1.1: bytes=56 Sequence=1 ttl=64 time=30 ms
    Reply from 10.1.1.1: bytes=56 Sequence=2 ttl=64 time=30 ms
    Reply from 10.1.1.1: bytes=56 Sequence=3 ttl=64 time=30 ms
    Reply from 10.1.1.1: bytes=56 Sequence=4 ttl=64 time=10 ms
    Reply from 10.1.1.1: bytes=56 Sequence=5 ttl=64 time=20 ms
  --- 10.1.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
0.00% packet loss
    round-trip min/avg/max = 10/24/30 ms
```

2. 配置 SSH Server

相比于 Telnet 协议，SSH 协议支持对报文加密传输，而非明文传送。因此，在跨越互联网的远程登录管理中，建议使用 SSH 协议。

由于SSH用户使用Password方式验证，需要在SSH服务器端生成本地RSA密钥，因此生成本地RSA密钥对是完成SSH登录配置的首要操作。

在R2上使用 **rsa local-key-pair create** 命令来生成本地RSA主机密钥对。

```
[R2]rsa local-key-pair create
The key name will be: R2_Host
The range of public key size is (512 ~ 2048).
NOTES: If the key modulus is greater than 512,
        it will take a few minutes.
Input the bits in the modulus[default = 512]:
Generating keys...
...................++++++++++++
................++++++++++++
.++++++++
................++++++++
```

配置完成后，使用 **display rsa local-key-pair public** 命令查看本地密钥对中的公钥部分信息。

```
[R2]display rsa local-key-pair public
=====================================================
Time of Key pair created: 2013-06-24 12:33:22-05:13
Key name: R2_Host
Key type: RSA encryption Key
=====================================================
Key code:
3047
  0240
    B7C2165E 055CE5B2 ACB91781 18996572 05AF6068
    F6B71A08 729D0494 84AD336D EAB8727C 2A8D4FB9
    DC0E2AE8 FAD182F6 37BF685B 7D730889 173FA1CE
    9621BF67
  0203
    010001
=====================================================
Time of Key pair created: 2013-06-24 12:33:27-05:13
Key name: Server
Key type: RSA encryption Key
=====================================================
Key code:
3067
  0260
    CEFA28DF E88B986F 8785B54E 035C0C4D 4671B975
    C71871E3 6F069F1C C1D7ACA2 DE279ED9 368EC812
    33E162D8 D03776C1 7757F05D A6D5F12E C5BBD88A
    40EDD70F 071E2E99 B5D7330C 26E3D393 BDDB3B98
    14E3086C 292F697D A973DC38 63C3570D
  0203
010001
```

可以观察到，此时已经生成了本地RSA主机密钥对。“Time of Key pair created”描述公钥生成的时间，“Key name”描述公钥的名称，“Key type”描述公钥的类型。

在R2上配置VTY用户界面，设置用户的验证方式为AAA授权验证方式。

```
[R2]user-interface vty 0 4
[R2-ui-vty0-4]authentication-mode aaa
```

指定VTY类型用户界面只支持SSH协议，设备将自动禁止Telnet功能。

```
[R2-ui-vty0-4]protocol inbound ssh
```

使用 **local-user** 命令创建本地用户和用户口令，并以密文方式显示用户口令，指定用户名为 huawei1，密码为 huawei1。

```
[R2]aaa
[R2-aaa]local-user huawei1 password cipher huawei1
Info: Add a new user.
```

配置本地用户的接入类型为 SSH。

```
[R2-aaa]local-user huawei1 service-type ssh
```

使用 **ssh user** 命令新建 SSH 用户，用户名为 huawei1，指定 SSH 用户的认证方式为 Password，即密码认证方式。

```
[R2]ssh user huawei1 authentication-type password
```

此处还可以继续使用 **local-user huawei1 privilege level** 命令配置本地用户的优先级。其取值范围为 0～15，取值越大，代表用户的优先级越高。不同级别的用户登录后，只能使用等于或低于自身级别的命令，默认值为 3，代表管理级。

默认情况下，设备的 SSH 服务器功能为关闭状态，只有开启了此功能后，用户端才能以 SSH 方式与设备建立连接。在 R2 上开启设备的 SSH 功能。

```
[R2]stelnet server enable
Info: Succeeded in starting the Stelnet server.
```

配置完成后，使用 **display ssh user-information huawei1** 命令在 SSH 服务器端查看 SSH 用户的配置信息。如果不在命令末尾指定 SSH 用户，则可以查看 SSH 服务器端所有的 SSH 用户配置信息。

```
[R2]display ssh user-information huawei1
----------------------------------------------------------------------------
 Username          Auth-type          User-public-key-name
----------------------------------------------------------------------------
huawei1           password            null
----------------------------------------------------------------------------
```

可以观察到所配置的 SSH 用户名及认证方式。

运行 **display ssh server status** 命令，可以查看 SSH 服务器全局配置信息。

```
[R2]display ssh server status
 SSH version                              :1.99
 SSH connection timeout                   :60 seconds
 SSH server key generating interval       :0 hours
 SSH Authentication retries               :3 times
 SFTP Server                              :Disable
Stelnet server                            :Enable
```

可以观察到，此时 R2 上 STelnet Server 服务器状态为启用状态。

3. 配置 SSH Client

当 SSH 用户端第一次登录 SSH 服务器时，用户端还没有保存 SSH 服务器的 RSA 公钥，会对服务器的 RSA 有效性公钥检查失败，从而导致登录服务器失败。因此当用户端 R1 首次登录时，需开启 SSH 用户端首次认证功能，不对 SSH 服务器的 RSA 公钥进行有效性检查。

```
[R1]ssh client first-time enable
```

在 SSH 用户端 R1 上使用 **stelnet** 命令连接 SSH 服务器。

```
[R1]stelnet 10.1.1.2
```

登录成功后，输入用户名 huawei1。

```
Please input the username:huawei1
```

```
Trying 10.1.1.2 ...
Press CTRL+K to abort
Connected to 10.1.1.2 ...

The server is not authenticated. Continue to access it? (y/n)[n]:y
Jun 24 2013 13:14:46-05:13 R1 %%01SSH/4/CONTINUE_KEYEXCHANGE(l)[0]:The server had not been authenticated in the process of exchanging keys. When deciding whether to continue, the user chose Y.
[R1]
Save the server's public key? (y/n)[n]:y
Jun 24 2013 1The server's public key will be saved with the name 10.1.1.2. Please wait...
3:14:50-05:13 R1 %%01SSH/4/SAVE_PUBLICKEY(l)[1]:When deciding whether to save the server's public key 10.1.1.2, the user chose Y.
```

第一次登录时，由于开启了 SSH 用户端首次认证功能，在 STelnet 用户端第一次登录 SSH 服务器时，将不对 SSH 服务器的 RSA 公钥进行有效性检查。登录后，系统将自动分配并保存 RSA 公钥，为下次登录时认证。

输入用户 huawei1 的密码 huawei1。

```
Enter password:
-------------------------------------------------------------------------------
  User last login information:
-------------------------------------------------------------------------------
  Access Type: SSH
IP-Address : 10.1.1.1 ssh
Time         : 2013-06-24 13:52:54-05:13
-------------------------------------------------------------------------------
<R2>
```

输入密码后，远程登录 R2 成功，使用 **display ssh server session** 命令查看 SSH 服务器端的当前会话连接信息。

```
[R2]display ssh server session
-------------------------------------------------------------------------------
 Conn    Ver    Encry       State   Auth-type            Username
-------------------------------------------------------------------------------
VTY 0   2.0    AES          run     password              huawei1
-------------------------------------------------------------------------------
```

可以观察到，用户 huawei1 已经成功通过 VTY 线路 0 远程登录上来，用户端已经成功连接到 SSH 服务器，可以进行各种配置。如果要退出登录，使用 **quit** 命令即可。

4. 配置 SFTP Server 与 Client

在 R2 上进入 AAA 视图，创建一个名称为 huawei2 的用户，并配置密码为 huawei2，以密文方式显示。

```
[R2]aaa
[R2-aaa]local-user huawei2 password cipher huawei2
```

配置本地用户的接入类型为 SSH。

```
[R2-aaa]local-user huawei2 service-type ssh
```

配置本地用户的优先级，不同级别的用户登录后，只能使用等于或低于自身级别的命令。取值范围为 0~15，取值越大，用户的优先级越高。

```
[R2-aaa]local-user huawei2 privilege level 3
```

指定 FTP 用户的可访问目录。默认为空，如果不配置，FTP 用户将无法登录。

```
[R2-aaa]local-user huawei2 ftp-directory flash:
```

使用 **ssh user** 命令新建 SSH 用户，用户名为 huawei2，指定 SSH 用户的认证方式为 Password，即密码认证方式。

```
[R2]ssh user huawei2 authentication-type password
```

使用 **sftp server enabe** 命令开启 SFTP 服务器功能。

```
[R2]sftp server enable
```

配置完成后，查看 SSH 服务器的配置信息。

```
[R2]display ssh server status
 SSH version                          :1.99
 SSH connection timeout               :60 seconds
 SSH server key generating interval   :0 hours
 SSH Authentication retries           :3 times
 SFTP Server                          :Enable
 Stelnet server                       :Enable
```

可以观察到，此时 SFTP 服务已经开启。

在 R1 上使用 **sftp** 命令连接 SSH 服务器，并输入用户名 huawei2 和口令 huawei2。

```
[R1]sftp 10.1.1.2
Please input the username:huawei2
Trying 10.1.1.2 ...
Press CTRL+K to abort
Enter password:
sftp-client>
```

可以观察到已经成功登录。

在 R2 上查看 SSH 会话连接信息。

```
[R2]display ssh server session
---------------------------------------------------------------------------
 Conn   Ver   Encry     State   Auth-type          Username
---------------------------------------------------------------------------
VTY 0   2.0   AES        run     password            huawei2
---------------------------------------------------------------------------
```

可以观察到，用户 huawei2 已经成功通过 VTY 线路 0 远程登录上来，用户端已经成功连接到 SSH 服务器，可以进行各种配置。如果要退出登录，使用 **quit** 命令即可。

思考

开启 SSH 用户端首次认证功能有什么缺陷？如果不开启此功能如何在用户端远程成功登录？

1.6 配置通过 FTP 进行文件操作

原理概述

FTP（File Transfer Protocol，文件传输协议）是在 TCP/IP 网络和 Internet 上最早使用的协议之一，在 TCP/IP 协议族中属于应用层协议，是文件传输的 Internet 标准。其主要功能是向用户提供本地和远程主机之间的文件传输，尤其是在进行版本升级、日志下载和配置保存等业务操作时。

FTP 采用 C/S（Client/Server）结构。FTP Server 能够提供远程用户端访问和操作的功能，用户可以通过主机或者其他设备上的 FTP 用户端程序登录到服务器上，进行文件的上传、下载和目录访问等操作。

实验目的

- 理解 FTP 的应用场景
- 掌握操作 FTP 服务器的常见命令
- 掌握保存文件到 FTP 的方法
- 掌握获取 FTP 服务器文件到本地的方法
- 掌握配置路由器为 FTP 服务器的方法

实验内容

本实验模拟企业网络。PC-1 为 FTP 用户端设备，需要访问 FTP Server，从服务器上下载或上传文件。出于安全角度考虑，为防止服务器被病毒文件感染，不允许用户端直接上传文件到 Server。网络管理员在 R1 上设置了限制，使员工不能上传文件到 Server，但是可以从 Server 下载文件。R1 也需要作为用户端从 Server 下载更新文件，同时配置 R1 作为 FTP 服务器，员工可上传文件到 R1 上，经过管理员的检测后由 R1 再上传到 FTP Server。

实验拓扑

配置通过 FTP 进行文件操作的拓扑如图 1-38 所示。

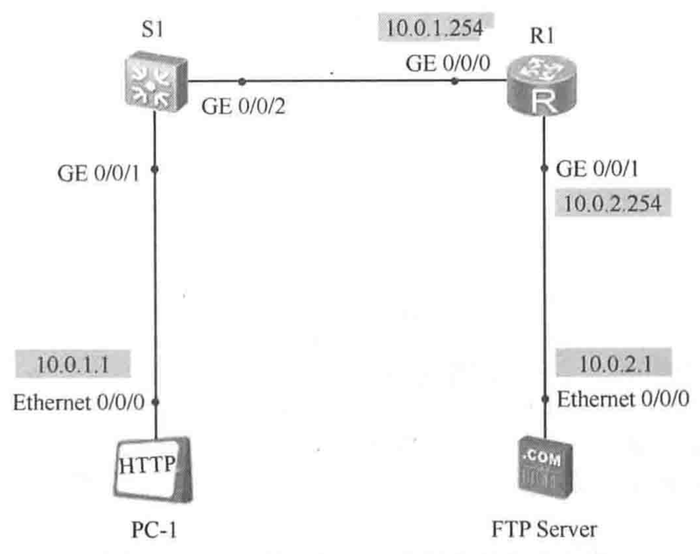


图 1-38　配置通过 FTP 进行文件操作拓扑

实验编址

实验编址见表 1-5。

表 1-5　　实验编址

设备	接口	IP 地址	子网掩码	默认网关
PC-1	Ethernet 0/0/0	10.0.1.1	255.255.255.0	10.0.1.254
FTP Server	Ethernet 0/0/0	10.0.2.1	255.255.255.0	10.0.2.254
R1（AR2220）	GE 0/0/0	10.0.1.254	255.255.255.0	N/A
	GE 0/0/1	10.0.2.254	255.255.255.0	N/A

实验步骤

1．基本配置

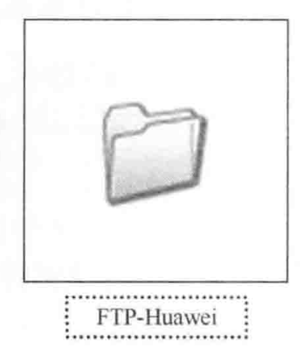
根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测各直连链路的连通性。

```
<R1>ping 10.0.1.1
PING 10.0.1.1: 56   data bytes, press CTRL_C to break
Reply from 10.0.1.1: bytes=56 Sequence=1 ttl=255 time=40 ms
Reply from 10.0.1.1: bytes=56 Sequence=2 ttl=255 time=10 ms
Reply from 10.0.1.1: bytes=56 Sequence=3 ttl=255 time=10 ms
Reply from 10.0.1.1: bytes=56 Sequence=4 ttl=255 time=1 ms
Reply from 10.0.1.1: bytes=56 Sequence=5 ttl=255 time=20 ms
--- 10.0.1.1 ping statistics ---
5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 1/16/40 ms
```

其余直连网段的连通性测试省略。

2．配置路由器为 FTP Client

首先，在本地电脑上创建一个文件夹 FTP-Huawei 作为 FTP 服务器的文件夹，在该文件夹下再创建子文件夹 Config，并创建测试文件 test.txt，如图 1-39 所示。



FTP-Huawei

图 1-39　创建文件夹

创建完成后，设置 FTP 服务器的文件夹为刚才的主文件夹目录，如图 1-40 所示。



图 1-40　设置 FTP 服务器文件根目录

设置完成后，启动 FTP Server。在 R1 上使用 **ftp** 命令连接 FTP 服务器。登录时默认需要输入用户名和密码，由于服务器上没有设置用户名和密码，每次在 R1 上输入时等同于创建该用户名和密码，本次使用用户名 10.0.2.1，密码 huawei。

```
<R1>ftp 10.0.2.1
Trying 10.0.2.1 ...
Press CTRL+K to abort
Connected to 10.0.2.1.
220 FtpServerTry FtpD for free
User(10.0.2.1:(none)):10.0.2.1
331 Password required for 10.0.2.1 .
Enter password:
230 User 10.0.2.1 logged in , proceed
[R1-ftp]
```

可以观察到，路由器进入 FTP 配置视图。

使用 **ls** 命令查看 FTP 服务器文件夹状态。

```
[R1-ftp]ls
200 Port command okay.
150 Opening ASCII NO-PRINT mode data connection for ls -l.
Config
226 Transfer finished successfully. Data connection closed.
FTP: 12 byte(s) received in 0.180 second(s) 66.66byte(s)/sec.
```

可以观察到，目前有文件夹 Config。

使用 **cd** 命令进入文件夹。

```
[R1-ftp]cd Config
250 "/config" is current directory.
```

可以观察到，目前已进入该文件夹。

使用 **dir** 命令查看详细的文件属性。

```
[R1-ftp]dir
200 Port command okay.
150 Opening ASCII NO-PRINT mode data connection for ls -l.
drwxrwxrwx   1 10.0.2.1   nogroup                  3 Aug 21   2013 test.txt
226 Transfer finished successfully. Data connection closed.
FTP: 66 byte(s) received in 0.050 second(s) 1.32Kbyte(s)/sec.
```

使用 **get** 命令下载 test.txt 到本地路由器。

```
[R1-ftp]get test.txt
200 Port command okay.
150 Sending test.txt (3 bytes). Mode STREAM Type BINARY
226 Transfer finished successfully. Data connection closed.
FTP: 3 byte(s) received in 17.450 second(s) .17byte(s)/s
```

可以观察到，下载文件成功。

使用 **put** 命令上传 test.txt 到 FTP 服务器，命名为 new.txt。

```
[R1-ftp]put test.txt new.txt
200 Port command okay.
150 Opening BINARY data connection for new.txt
226 Transfer finished successfully. Data connection closed.
FTP: 3 byte(s) sent in 0.070 second(s) 42.85byte(s)/sec.
```

可以观察到，上传文件成功。

3. 配置路由器为 FTP Server

在上面的步骤中，路由器作为 FTP Client 已经成功从 FTP Server 上获取和上传了文件。

现在将路由器配置为 FTP 服务器，可以使得路由器下行的用户端能够上传文件到路由器上，并可直接从 Server 上获取文件。

打开路由器 R1 的 FTP 服务器功能。

```
<R1>system-view
[R1]ftp server enable
```

设置 FTP 登录的用户名为 ftp，密码为 huawei，设置文件夹目录“flash:”。配置 FTP 用户可访问的目录为“fash:”，用户优先级为 15，服务类型为 ftp。

```
[R1]aaa
[R1-aaa]local-user ftp password cipher hawei
[R1-aaa]local-user ftp ftp-directory flash:
[R1-aaa]local-user ftp service-type ftp
[R1-aaa]local-user ftp privilege level 15
```

配置完成后，在本地创建测试文件 test-user.txt，并设置用户端信息如图 1-41 所示。配置服务器地址为 10.0.1.254，用户名为 ftp，密码为 huawei，然后单击“登录”按钮。



图 1-41 在 PC-1 上设置 FTP 服务器

登录成功后，可在“本地文件列表”中选择文件 test-user.txt，并单击向右箭头传送至 FTP 服务器，可观察到上传文件成功。

在 R1 上查看目录下的文件。

```
[R1]dir
Directory of flash:/
  Idx   Attr     Size(Byte)  Date        Time(LMT)  FileName
```

```
......
3   -rw-             0   Sep 09 2013 03:32:58    test-user.txt
    4   -rw-             0   Sep 09 2013 03:25:47    test.txt
980,052 KB total (700,320 KB free)
```

可以观察到，已经将相应文件成功上传至 FTP 服务器 R1。

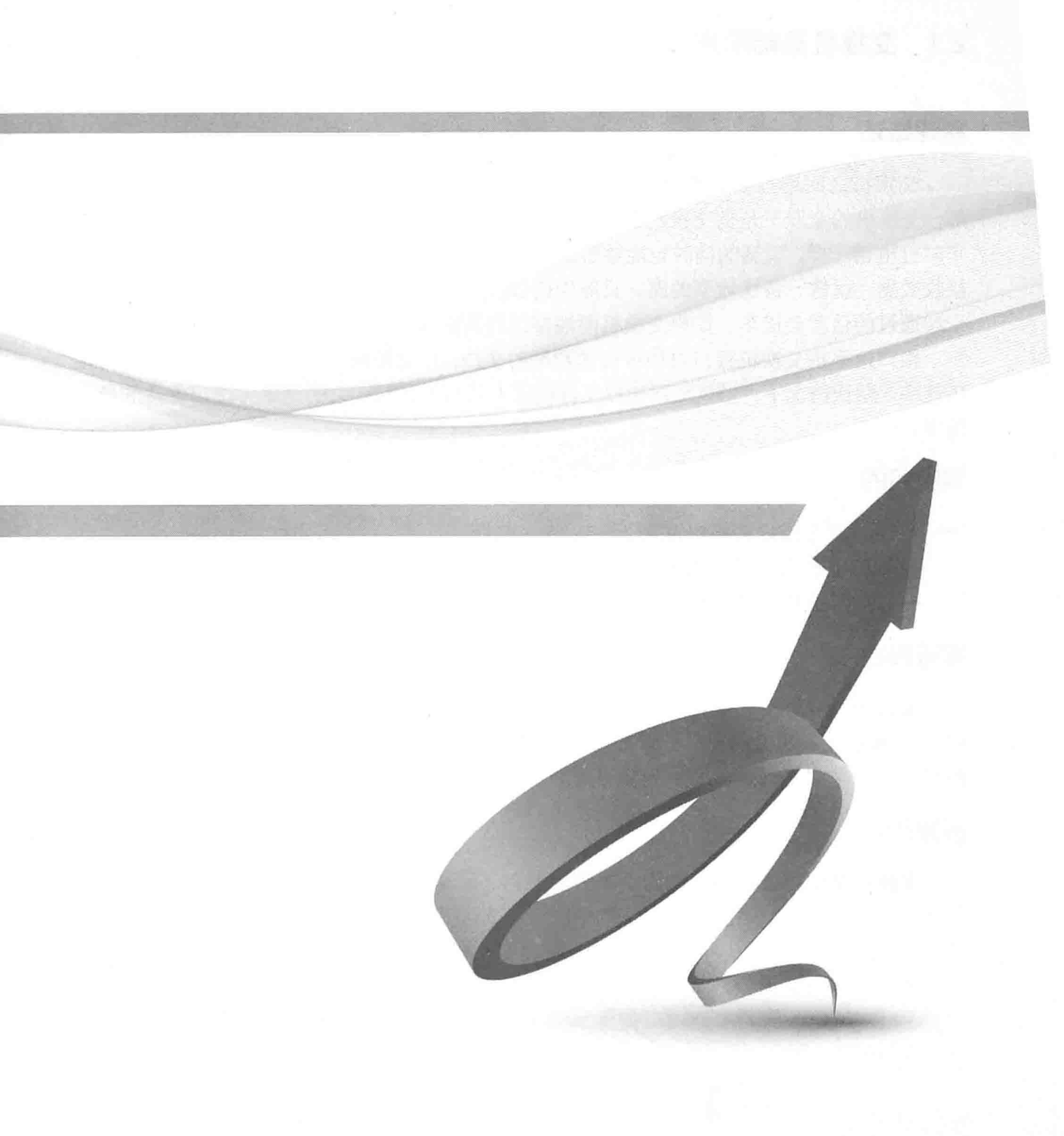
思考

默认情况下，FTP 服务器端监听端口号是 21，能否在路由器上变更此端口号？有什么好处？

第2章 交换机基础配置

2.1 交换机基础配置

2.2 理解ARP及Proxy ARP



2.1 交换机基础配置

原理概述

交换机之间通过以太网电接口对接时需要协商一些接口参数，比如速率、双工模式等。交换机的全双工是指交换机在发送数据的同时也能够接收数据，两者同时进行。就如平时打电话一样，说话的同时也能够听到对方的声音。而半双工指在同一时刻只能发送或接收数据，就像一条比较窄的路，只能先通过一边的车，然后再通过另一边的车，若两边一起通过的话就会撞车。如果交换机两端接口协商模式不一致，会导致报文交互异常。

接口速率指交换机接口每秒钟传输数据的多少，在交换机上可根据需要调整以太网接口速率。默认情况下，当以太网接口工作在非自协商模式时，它的速率为接口支持的最大速率。

实验目的

- 理解双工模式和接口速率
- 掌握更改双工模式的配置
- 掌握更改接口速率的配置

实验内容

某公司刚成立，新组建网络，购置了 3 台交换机。其中 S1 和 S2 为接入层交换机，S3 为汇聚层交换机。现在网络管理员需要对 3 台新交换机进行基本配置，保证交换机间的接口使用全双工模式，并根据需要配置接口速率。

实验拓扑

交换机基础配置的拓扑如图 2-1 所示。

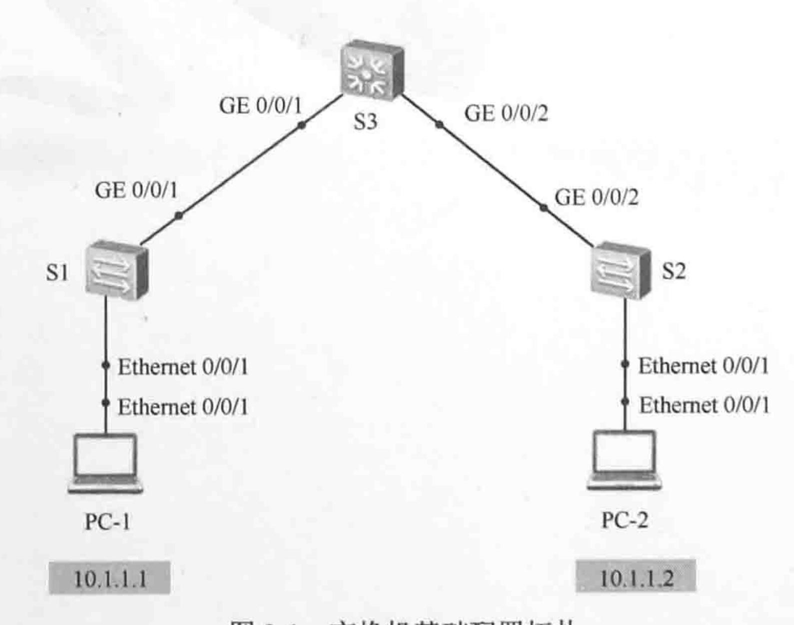


图 2-1 交换机基础配置拓扑

实验编址

实验编址见表 2-1。

表 2-1 实验编址

设备	接口	IP 地址	子网掩码	默认网关
PC-1	Ethernet 0/0/1	10.1.1.1	255.255.255.0	N/A
PC-2	Ethernet 0/0/1	10.1.1.2	255.255.255.0	N/A

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 地址配置，并使用 **ping** 命令检测各直连链路的连通性。

```
PC>ping 10.1.1.2
Ping 10.1.1.2: 32 data bytes, Press Ctrl_C to break
From 10.1.1.2: bytes=32 seq=1 ttl=128 time=62 ms
From 10.1.1.2: bytes=32 seq=2 ttl=128 time=63 ms
From 10.1.1.2: bytes=32 seq=3 ttl=128 time=31 ms
From 10.1.1.2: bytes=32 seq=4 ttl=128 time=47 ms
From 10.1.1.2: bytes=32 seq=5 ttl=128 time=78 ms
--- 10.1.1.2 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
0.00% packet loss
  round-trip min/avg/max = 31/56/78 ms
```

现在 PC-1 和 PC-2 能够正常通信。

2. 配置交换机双工模式

配置接口的双工模式可在自协商或者非自协商模式下进行。

在自协商模式下，接口的双工模式是和对端接口协商得到的，但协商得到的双工模式可能与实际要求不符。可通过配置双工模式的取值范围来控制协商的结果。例如，互连的两个设备对应的接口都支持全/半双工，经自协商后工作在半双工模式，与实际要求的全双工模式不符，这时就可以执行 **auto duplex full** 命令使接口的可协商双工模式变为全双工模式。默认情况下，以太网接口自协商双工模式范围为接口所支持的双工模式。

在非自协商模式下，可以根据实际需求手动配置接口的双工模式。

在 S1、S2、S3 交换机接口下先通过 **undo negotiation auto** 命令关掉自协商功能，再手工指定双工模式为全双工。

```
[S1]interface GigabitEthernet 0/0/1
[S1-GigabitEthernet0/0/1]undo negotiation auto
[S1-GigabitEthernet0/0/1]duplex full

[S2]interface GigabitEthernet 0/0/2
[S2-GigabitEthernet0/0/2]undo negotiation auto
[S2-GigabitEthernet0/0/2]duplex full

[S3]interface GigabitEthernet 0/0/1
[S3-GigabitEthernet0/0/1]undo negotiation auto
[S3-GigabitEthernet0/0/1]duplex full
```

```
[S3-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2
[S3-GigabitEthernet0/0/2]undo negotiation auto
[S3-GigabitEthernet0/0/2]duplex full
```

3. 配置接口速率

在自协商模式下，以太网接口的速率是和对端接口协商得到的。如果协商的速率与实际要求不符，可通过配置速率的取值范围来控制协商的结果。例如，互连的两个设备对应的接口经自协商后的速率为 10Mbit/s，与实际要求的 100Mbit/s 不符，可通过执行 **auto speed 100** 命令配置使得接口可协商的速率为 100Mbit/s。默认情况下，以太网接口自协商速率范围为接口支持的所有速率。

在非自协商模式下，需手动配置接口速率，避免发生无法正常通信的情况。

默认情况下，以太网接口的速率为接口支持的最大速率。

根据网络需要调整接口速率。由于网络用户较少，配置 GE 接口速率为 100Mbit/s，配置 Ethernet 接口速率为 10Mbit/s。

在 3 台交换机接口下配置速率。首先关闭接口自协商模式，然后配置以太网接口的速率。

```
[S1]interface Ethernet 0/0/1
[S1-Ethernet0/0/1]undo negotiation auto
[S1-Ethernet0/0/1]speed 10
```

用同样的方法配置另外两台设备接口的速率。

```
[S2]interface Ethernet 0/0/1
[S2-Ethernet0/0/1]undo negotiation auto
[S2-Ethernet0/0/1]speed 10
[S2-Ethernet0/0/1]interface GigabitEthernet 0/0/2
[S2-GigabitEthernet0/0/2]undo negotiation auto
[S2-GigabitEthernet0/0/2]speed 100

[S3]interface GigabitEthernet 0/0/1
[S3-GigabitEthernet0/0/1]undo negotiation auto
[S3-GigabitEthernet0/0/1]speed 100
[S3-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2
[S3-GigabitEthernet0/0/2]undo negotiation auto
[S3-GigabitEthernet0/0/2]speed 100
```

思考

在实际操作中，通常使用自动协商模式还是手动配置模式？为什么？

2.2 理解 ARP 及 Proxy ARP

原理概述

ARP（Address Resolution Protocol）是用来将 IP 地址解析为 MAC 地址的协议。ARP 表项可以分为动态和静态两种类型。动态 ARP 是利用 ARP 广播报文，动态执行并自动进行 IP 地址到以太网 MAC 地址的解析，无需网络管理员手工处理。静态 ARP 是建立 IP 地址和 MAC 地址之间固定的映射关系，在主机和路由器上不能动态调整此映射关系，需要网络管理员手工添加。设备上有一个 ARP 高速缓存（ARP cache），用来存放 IP 地

址到 MAC 地址的映射表，利用 ARP 请求和应答报文来缓存映射表，以便能正确地把三层数据包封装成二层数据帧，达到快速封装数据帧、正确转发数据的目的。另外 ARP 还有扩展应用功能，比如 Proxy ARP 功能。

Proxy ARP，即代理 ARP，当主机上没有配置默认网关地址（即不知道如何到达本地网络的网关设备），可以发送一个广播 ARP 请求（请求目的主机的 MAC 地址），使具备 Proxy ARP 功能的路由器收到这样的请求后，在确认请求地址可达后，会使用自身的 MAC 地址作为该 ARP 请求的回应，使得处于不同物理网络的同一网段的主机之间可以正常通信。

实验目的

- 理解 ARP 工作原理
- 掌握配置静态 ARP 的方法
- 理解 Proxy ARP 的工作原理
- 掌握 Proxy ARP 的配置
- 理解主机之间的通信过程

实验内容

本实验模拟公司网络场景。路由器 R1 是公司的出口网关，连接到外网。公司内所有员工使用 10.1.0.0/16 网段，通过交换机连接到网关路由器上。网络管理员通过配置静态 ARP 防止 ARP 欺骗攻击，保证通信安全。又由于公司内所有主机都没有配置网关，且分属于不同广播域，造成无法正常通信，网络管理员需要通过在路由器上配置 ARP 代理功能，实现网络内所有主机的通信。

实验拓扑

ARP 及 ARP Proxy 的拓扑如图 2-2 所示。

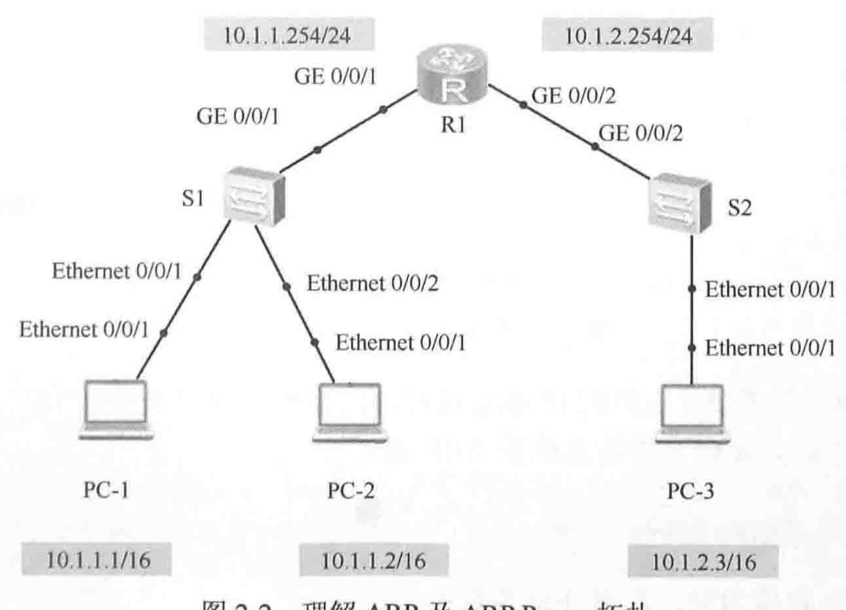


图 2-2　理解 ARP 及 ARP Proxy 拓扑

实验编址

实验编址见表 2-2。

表 2-2 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR2220）	GE 0/0/1	10.1.1.254	255.255.255.0	N/A
	GE 0/0/2	10.1.2.254	255.255.255.0	N/A
PC-1	Ethernet 0/0/1	10.1.1.1	255.255.0.0	N/A
PC-2	Ethernet 0/0/1	10.1.1.2	255.255.0.0	N/A
PC-3	Ethernet 0/0/1	10.1.2.3	255.255.0.0	N/A

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 地址配置，并使用 **ping** 命令检测各直连链路的连通性，如图 2-3 所示。



图 2-3 配置 PC-1 的 IP 地址

根据实验编址配置 PC 主机的 IP 地址及对应的掩码，掩码长度是 16。配置完成后，在命令行下使用 **arp –a** 命令查看主机的 ARP 表。

```
PC>arp -a
Internet Address     Physical Address     Type
PC>
```

查看到 ARP 表项为空，没有任何条目在里面。

根据实验编址配置路由器 R1 的接口 IP 地址，掩码长度为 24。

```
[R1]interface GigabitEthernet 0/0/1
[R1-GigabitEthernet0/0/1]ip address 10.1.1.254 255.255.255.0
[R1-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2
[R1-GigabitEthernet0/0/2]ip address 10.1.2.254 255.255.255.0
```

配置完成后，使用 **display arp all** 命令查看 R1 的 ARP 表。

```
[R1]display arp all
IP ADDRESS      MAC ADDRESS     EXPIRE(M) TYPE        INTERFACE   VPN-INSTANCE
VLAN/CEVLAN PVC
------------------------------------------------------------------------------
10.1.1.254      00e0-fc03-beac            I -         GE0/0/1
10.1.2.254      00e0-fc03-bead            I -         GE0/0/2
------------------------------------------------------------------------------
Total:2         Dynamic:0       Static:0    Interface:2
```

ARP 表中仅含有 R1 的两个接口 IP 地址及与其对应的 MAC 地址的 ARP 表项，没有其他条目。

在 PC-1 上使用 **ping** 命令测试到网关 R1 和 PC-2 的连通性。

```
PC>ping 10.1.1.254
Ping 10.1.1.254: 32 data bytes, Press Ctrl_C to break
From 10.1.1.254: bytes=32 seq=1 ttl=255 time=47 ms
From 10.1.1.254: bytes=32 seq=2 ttl=255 time=47 ms
From 10.1.1.254: bytes=32 seq=3 ttl=255 time=31 ms
From 10.1.1.254: bytes=32 seq=4 ttl=255 time=31 ms
From 10.1.1.254: bytes=32 seq=5 ttl=255 time=15 ms
--- 10.1.1.254 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 15/34/47 ms

PC>ping 10.1.1.2
Ping 10.1.1.2: 32 data bytes, Press Ctrl_C to break
From 10.1.1.2: bytes=32 seq=1 ttl=128 time=16 ms
From 10.1.1.2: bytes=32 seq=2 ttl=128 time<1 ms
From 10.1.1.2: bytes=32 seq=3 ttl=128 time=16 ms
From 10.1.1.2: bytes=32 seq=4 ttl=128 time=16 ms
From 10.1.1.2: bytes=32 seq=5 ttl=128 time=16 ms
--- 10.1.1.2 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 0/12/16 ms
```

在 PC-3 上，使用 **ping** 命令测试到网关 R1 的连通性。

```
PC>ping 10.1.2.254
Ping 10.1.2.254: 32 data bytes, Press Ctrl_C to break
From 10.1.2.254: bytes=32 seq=1 ttl=255 time=47 ms
From 10.1.2.254: bytes=32 seq=2 ttl=255 time=46 ms
From 10.1.2.254: bytes=32 seq=3 ttl=255 time=31 ms
From 10.1.2.254: bytes=32 seq=4 ttl=255 time=16 ms
From 10.1.2.254: bytes=32 seq=5 ttl=255 time=15 ms
--- 10.1.2.254 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
```

```
0.00% packet loss
round-trip min/avg/max = 15/31/47 ms
```

可以观察到，直连网络连通性正常。

当主机和网关之间有数据访问时，如果 ARP 表中没有目标 IP 地址与目标 MAC 地址的对应表项，ARP 协议会被触发，向直连网段发送 ARP 广播请求包，请求目标 IP 地址所对应的 MAC 地址。图 2-4 是 PC-1 发送的 ARP 广播请求，请求目标 IP 10.1.1.254 的 MAC 地址。

```
⊞ Ethernet II, Src: HuaweiTe_cf:26:03 (54:89:98:cf:26:03), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
⊟ Address Resolution Protocol (request)
    Hardware type: Ethernet (0x0001)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (0x0001)
    [Is gratuitous: False]
    Sender MAC address: HuaweiTe_cf:26:03 (54:89:98:cf:26:03)
    Sender IP address: 10.1.1.1 (10.1.1.1)
    Target MAC address: Broadcast (ff:ff:ff:ff:ff:ff)
    Target IP address: 10.1.1.254 (10.1.1.254)
```

图 2-4 PC-1 发送的 ARP 广播请求报文

网关收到广播请求后，回应单播的 ARP 响应，里面含有自身 IP 地址与 MAC 地址的对应关系，如图 2-5 所示。

```
⊞ Ethernet II, Src: HuaweiTe_03:be:ac (00:e0:fc:03:be:ac), Dst: HuaweiTe_cf:26:03 (54:89:98:cf:26:03)
⊟ Address Resolution Protocol (reply)
    Hardware type: Ethernet (0x0001)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (0x0002)
    [Is gratuitous: False]
    Sender MAC address: HuaweiTe_03:be:ac (00:e0:fc:03:be:ac)
    Sender IP address: 10.1.1.254 (10.1.1.254)
    Target MAC address: HuaweiTe_cf:26:03 (54:89:98:cf:26:03)
    Target IP address: 10.1.1.1 (10.1.1.1)
```

图 2-5 ARP 响应报文

PC 和 R1 都会从这一对消息中知道对方的 IP 地址与 MAC 地址的对应关系，并将它写到各自的 ARP 表中。在 PC-1 上 **ping** 网关 10.1.1.254 后，在 PC 上使用 **arp–a** 命令查看，在 R1 上使用 **display arp all** 命令查看。

```
PC>arp -a
Internet Address      Physical Address      Type
10.1.1.254              00e0-fc03-beac      dynamic

<R1>display arp all
IP ADDRESS      MAC ADDRESS      EXPIRE(M) TYPE        INTERFACE    VPN-INSTANCE
VLAN/CEVLAN PVC
------------------------------------------------------------------------------
10.1.1.254      00e0-fc03-beac                  I -        GE0/0/1
10.1.1.1        5489-98cf-2603          20      D-0        GE0/0/1
10.1.2.254      00e0-fc03-bead                  I -        GE0/0/2
10.1.2.3        5489-98cf-e417          20      D-0        GE0/0/2
------------------------------------------------------------------------------
Total:4         Dynamic:2       Static:0      Interface:2
```

可以观察到，在 PC-1 上生成了网关 IP 地址 10.1.1.254 和与其对应的 MAC 地址的 ARP 表项，在 R1 上生成了 PC-1 的 IP 地址 10.1.1.1 和与其对应的 MAC 地址的 ARP 表项。上述出现在 PC 和 R1 里面的条目都是动态生成的。如果一段时间之后没有更新，便会从上述 ARP 表中删除。

2. 配置静态 ARP

上述 ARP 协议的工作行为往往被攻击者利用。如果攻击者发送伪造的 ARP 报文，而且报文里面所通告的 IP 地址和 MAC 地址的映射是错误的，则主机或网关会把错误的映射更新到 ARP 表中。当主机要发送数据到指定的目标 IP 地址时，从 ARP 表里得到了不正确的硬件 MAC 地址，并用之封装数据帧，导致数据帧无法正确发送。

由于公司内主机感染了这种 ARP 病毒，所以主机对网关 R1 进行 ARP 攻击，向网关 R1 通告含错误映射的 ARP 通告，导致网关路由器上使用不正确的动态 ARP 映射条目，造成其他主机无法与网关正常通信。

模拟 ARP 攻击发生时，网络的通信受到了影响。在网关 R1 上，使用 **arp static 10.1.1.1 5489-98CF-2803** 命令在路由器上静态添加一条关于 PC-1 的错误的 ARP 映射，假定此映射条目是通过一个 ARP 攻击报文所获得的（静态的条目优于动态的条目），所以错误的映射将出现在 ARP 表中。

```
[R1]arp static 10.1.1.1 5489-98CF-2803
```

使用 **display arp all** 命令查看 ARP 表，并使用 **ping** 测试 PC-1 和网关间的连通性。

```
[R1]display arp all
IP ADDRESS      MAC ADDRESS     EXPIRE(M) TYPE      INTERFACE   VPN-INSTANCE
VLAN/CEVLAN PVC
------------------------------------------------------------------------------
10.1.1.254      00e0-fc03-beac            I -       GE0/0/1
10.1.2.254      00e0-fc03-bead            I -       GE0/0/2
10.1.1.1        5489-98cf-2803            S--
------------------------------------------------------------------------------
Total:3         Dynamic:0       Static:1      Interface:2

[R1]ping 10.1.1.1
PING 10.1.1.1: 56  data bytes, press CTRL_C to break
Request time out
Request time out
Request time out
Request time out
Request time out
......

PC>ping 10.1.1.254
Ping 10.1.1.254: 32 data bytes, Press Ctrl_C to break
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
......
```

可以观察到，PC-1 与网关间无法通信，因为在路由器 R1 上 ARP 的映射错误，导致路由器无法正确地发送数据包给 PC-1。在 R1 的 GE 0/0/1 接口抓包观察，如

图 2-6 所示。

```
 8 14.024000 10.1.1.254  10.1.1.1  ICMP  Echo (ping) request  (id=0xcdab, seq(be/le)=256/1, ttl=255)
10 16.037000 10.1.1.254  10.1.1.1  ICMP  Echo (ping) request  (id=0xcdab, seq(be/le)=512/2, ttl=255)
12 18.049000 10.1.1.254  10.1.1.1  ICMP  Echo (ping) request  (id=0xcdab, seq(be/le)=768/3, ttl=255)
14 20.077000 10.1.1.254  10.1.1.1  ICMP  Echo (ping) request  (id=0xcdab, seq(be/le)=1024/4, ttl=255)
16 22.090000 10.1.1.254  10.1.1.1  ICMP  Echo (ping) request  (id=0xcdab, seq(be/le)=1280/5, ttl=255)

⊞ Frame 14: 98 bytes on wire (784 bits), 98 bytes captured (784 bits)
⊟ Ethernet II, Src: HuaweiTe_03:be:ac (00:e0:fc:03:be:ac), Dst: HuaweiTe_cf:28:03 (54:89:98:cf:28:03)
  ⊞ Destination: HuaweiTe_cf:28:03 (54:89:98:cf:28:03)
  ⊞ Source: HuaweiTe_03:be:ac (00:e0:fc:03:be:ac)
    Type: IP (0x0800)
⊞ Internet Protocol, Src: 10.1.1.254 (10.1.1.254), Dst: 10.1.1.1 (10.1.1.1)
⊞ Internet Control Message Protocol
```

图 2-6　抓包现象

可以观察到，由于配置了静态 ARP，R1 发往 PC-1 的 **ping** 包的二层头部，目的 MAC 地址被错误地封装为 5489-98CF-2803。

应对 ARP 欺骗攻击，防止其感染路由器的 ARP 表，可以通过配置静态 ARP 表项来实现。如果 IP 地址和一个 MAC 地址的静态映射已经出现在 ARP 表中，则通过动态 ARP 方式学来的映射则无法进入 ARP 表。所以针对公司网络的现状，网络管理员在 R1 上手工配置三条关于 PC-1、PC-2 和 PC-3 的正确 ARP 映射。使用 **arp static** 命令，配置如下。

```
[R1]undo arp static 10.1.1.1 5489-98cf-2803
[R1]arp static 10.1.1.1 5489-98cf-2603
[R1]arp static 10.1.1.2 5489-98cf-5723
[R1]arp static 10.1.2.3 5489-98cf-e417
[R1]display arp all
IP ADDRESS      MAC ADDRESS     EXPIRE(M) TYPE        INTERFACE    VPN-INSTANCE
VLAN/CEVLAN PVC
------------------------------------------------------------------------------
10.1.1.254      00e0-fc03-beac            I -         GE0/0/1
10.1.2.254      00e0-fc03-bead            I -         GE0/0/2
10.1.1.1        5489-98cf-2603            S--
10.1.1.2        5489-98cf-5723            S--
10.1.2.3        5489-98cf-e417            S--
------------------------------------------------------------------------------
Total:5         Dynamic:0       Static:3        Interface:2
```

配置完成后，在 PC-1 上测试。

```
PC>ping 10.1.1.254
Ping 10.1.1.254: 32 data bytes, Press Ctrl_C to break
From 10.1.1.254: bytes=32 seq=1 ttl=255 time=31 ms
From 10.1.1.254: bytes=32 seq=2 ttl=255 time=31 ms
From 10.1.1.254: bytes=32 seq=3 ttl=255 time=31 ms
From 10.1.1.254: bytes=32 seq=4 ttl=255 time=16 ms
From 10.1.1.254: bytes=32 seq=5 ttl=255 time=16 ms
--- 10.1.1.254 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 16/25/31 ms
```

可以观察到，配置后连通性恢复正常。

公司网络中出现 ARP 攻击的情况是比较常见的，防御的办法之一是在 ARP 表中手工添加 ARP 映射。此种方法的优点是简单易操作，不足之处是网络中每个网络设备都有

ARP 表，要全方位保护网络就要在尽可能多的三层设备上把全网的 ARP 映射手工写入到 ARP 表里，工作量过大，如果更换 IP 或 MAC 后，还需要手工更新 ARP 映射，远没有动态 ARP 协议维护 ARP 表方便。但如果公司网络规模不大或者网络设备不多的情况下，静态 ARP 方案还是具有一定优势的。

3. 配置 Proxy ARP

目前公司的网络被路由器 R1 分割为两个独立的广播域，每个路由器接口对应一个 IP 网络，分别是 10.1.1.0/24 和 10.1.2.0/24，查看 R1 的路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 6        Routes : 6
Destination/Mask    Proto   Pre  Cos     Flags  NextHop        Interface
 10.1.1.0/24        Direct  0    0       D      10.1.1.254     GigabitEthernet 0/0/1
 10.1.1.254/32      Direct  0    0       D      127.0.0.1      GigabitEthernet 0/0/1
 10.1.2.0/24        Direct  0    0       D      10.1.2.254     GigabitEthernet 0/0/2
 10.1.2.254/32      Direct  0    0       D      127.0.0.1      GigabitEthernet 0/0/2
 127.0.0.0/8        Direct  0    0       D      127.0.0.1      InLoopBack0
 127.0.0.1/32       Direct  0    0       D      127.0.0.1      InLoopBack0
```

默认情况下，路由器上的 ARP 代理功能是关闭的。

如果 R1 保持 ARP 代理功能关闭的情况，则 PC-2 和 PC-3 之间不能互相通信。在 PC-2 上，使用 **ping** 访问 10.1.2.3，并在 PC-2 的 E 0/0/1 接口上抓包来观察，如图 2-7 所示。

```
PC>ping 10.1.2.3
Ping 10.1.2.3: 32 data bytes, Press Ctrl_C to break
From 10.1.1.2: Destination host unreachable
From 10.1.1.2: Destination host unreachable
From 10.1.1.2: Destination host unreachable
From 10.1.1.2: Destination host unreachable
From 10.1.1.2: Destination host unreachable
……
```

No.	Time	Source	Destination	Protocol	Info
344	690.741000	HuaweiTe_cf:57:23	Broadcast	ARP	Who has 10.1.2.3? Tell 10.1.1.2
345	691.755000	HuaweiTe_cf:57:23	Broadcast	ARP	Who has 10.1.2.3? Tell 10.1.1.2
347	692.754000	HuaweiTe_cf:57:23	Broadcast	ARP	Who has 10.1.2.3? Tell 10.1.1.2

图 2-7 抓包观察

可以观察到，PC-2 发出了 ARP 广播，却一直没有收到 ARP 响应。原因是 PC-2 和 PC-3 分处在两个广播域内，PC-2 发的 ARP 请求无法跨越中间的路由器，所以 PC-3 收不到 PC-2 的 ARP 请求，PC-2 也无法知晓目标主机 PC-3 的硬件 MAC 地址而导致数据封装失败。

但 R1 如果开启 ARP 代理之后，看是否能够解决这个问题。配置 **arp-proxy enable** 命令在路由器的接口上来开启 ARP 代理功能。

```
[R1]interface GigabitEthernet 0/0/1
[R1-GigabitEthernet0/0/1]arp-proxy enable
```

启用 ARP 代理之后，同样的测试，在 PC-2 上访问 PC-3，并在 PC-2 的 E 0/0/1 接口抓包观察，如图 2-8 所示。

```
No.   Time       Source              Destination         Protocol Info
    4 4.712000   HuaweiTe_cf:57:23   Broadcast           ARP      Who has 10.1.2.3?  Tell 10.1.1.2
    5 4.743000   HuaweiTe_03:be:ac   HuaweiTe_cf:57:23   ARP      10.1.2.3 is at 00:e0:fc:03:be:ac

⊞ Frame 5: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
⊞ Ethernet II, Src: HuaweiTe_03:be:ac (00:e0:fc:03:be:ac), Dst: HuaweiTe_cf:57:23 (54:89:98:cf:57:23)
⊟ Address Resolution Protocol (reply)
    Hardware type: Ethernet (0x0001)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (0x0002)
    [Is gratuitous: False]
    Sender MAC address: HuaweiTe_03:be:ac (00:e0:fc:03:be:ac)
    Sender IP address: 10.1.2.3 (10.1.2.3)
    Target MAC address: HuaweiTe_cf:57:23 (54:89:98:cf:57:23)
    Target IP address: 10.1.1.2 (10.1.1.2)
```

图 2-8 抓包现象

可以观察到，PC-2 发出了 ARP 请求并收到了 ARP 响应，但响应中 10.1.2.3 所对应的硬件 MAC 地址并非是 PC-3 的 MAC 地址，而是网关 R1 的 GE 0/0/1 接口 MAC 地址。

在 PC-2 上查看 ARP 表。

```
PC>arp -a
Internet Address    Physical Address    Type
10.1.2.3            00-e0-fc-03-be-ac   dynamic
```

开启 ARP 代理后，PC-2 访问 PC-3 的工作过程如下。

R1 的接口 GE 0/0/1 开启了 ARP 代理后，收到 PC-2 的 ARP 广播请求报文后，R1 根据 ARP 请求中的目标 IP 地址 10.1.2.3 查看自身的路由表中是否有对应的目标网络，R1 的 GE 0/0/2 接口就是 10.1.2.0/24 网络，所以，R1 直接把自身的 GE 0/0/1 接口的 MAC 地址通过 ARP 响应返回给 PC-2，PC-2 接收到此 ARP 响应后使用该 MAC 作为目标硬件地址发送报文给 R1，R1 收到后再把报文转发给 PC-3。

同理，PC-3 要能访问 R1 连接的其他广播域的 PC，也需要在 R1 的 GE 0/0/2 接口上开启 ARP 代理功能。

```
[R1]interface GigabitEthernet 0/0/2
[R1-GigabitEthernet0/0/2]arp-proxy enable
```

配置完成后，测试 PC-3 与 PC-1 间的连通性。

```
PC>ping 10.1.1.1
Ping 10.1.1.1: 32 data bytes, Press Ctrl_C to break
From 10.1.1.1: bytes=32 seq=1 ttl=127 time=47 ms
From 10.1.1.1: bytes=32 seq=2 ttl=127 time=47 ms
From 10.1.1.1: bytes=32 seq=3 ttl=127 time=62 ms
From 10.1.1.1: bytes=32 seq=4 ttl=127 time=62 ms
From 10.1.1.1: bytes=32 seq=5 ttl=127 time=78 ms
--- 10.1.1.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 47/59/78 ms
```

可以观察到，通信正常。

如果 IP 网络过大，广播对网络的影响也相应增大。在不改变网络主机配置的情况下，由管理员在网络中透明地插入一台路由器，靠路由器分割出多个广播域，降低了广播对网络的影响。在当前的 IP 网络中，此种做法并不多见。其缺点是主机间的通信

会因为引入额外的路由器而延迟增大，并存在着瓶颈问题，所以一般只作为临时解决方案使用。

思考

在 ARP 代理开启的情况下，如果在 PC-2 上，**ping** 10.1.2.4（10.1.2.4 主机不存在），icmp echo 报文是在 PC-2 还是 R1 路由器丢掉的？为什么？

第3章 VLAN

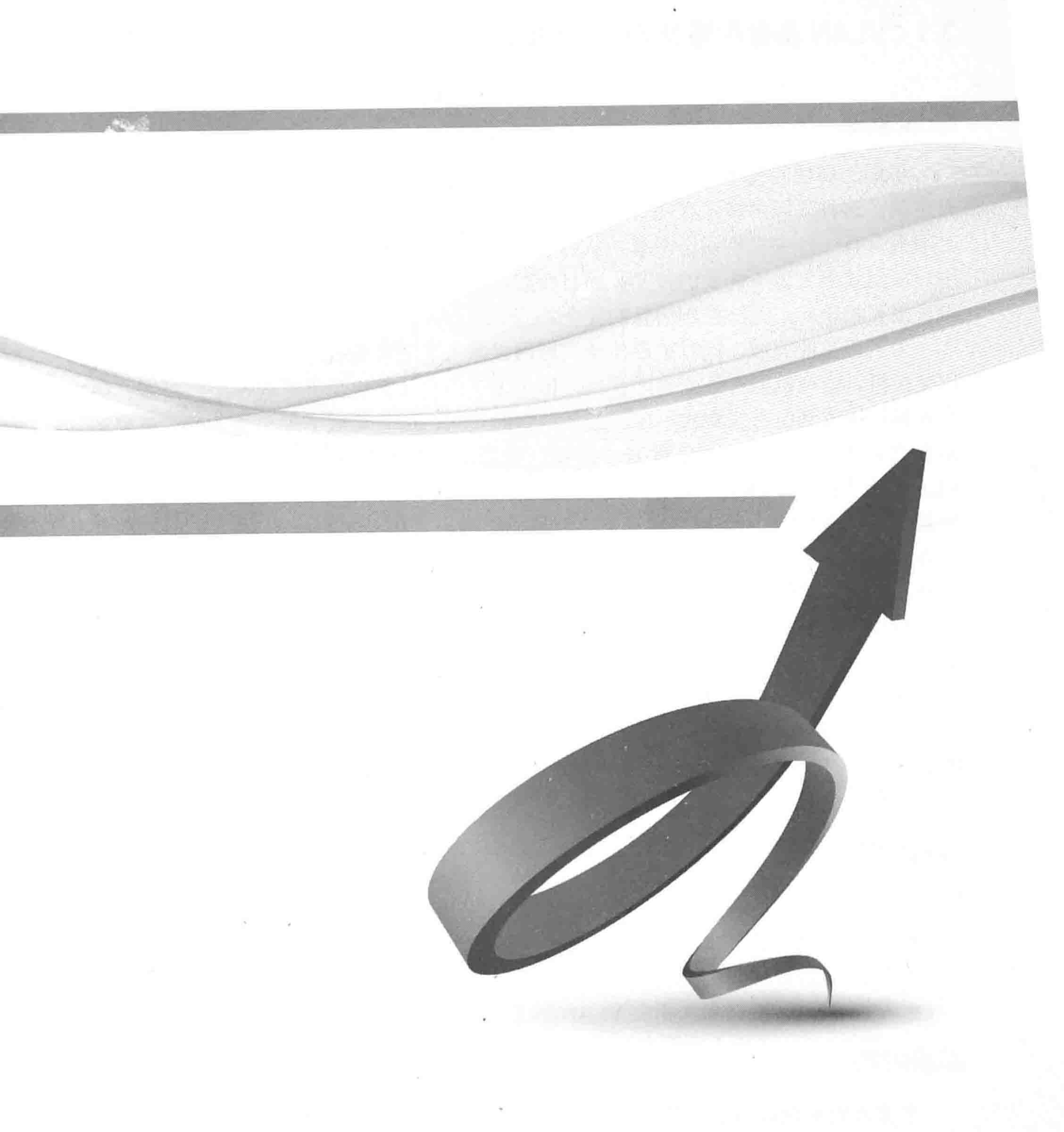
3.1 VLAN基础配置及Access接口

3.2 配置Trunk接口

3.3 理解Hybrid接口的应用

3.4 利用单臂路由实现VLAN间路由

3.5 利用三层交换机实现VLAN间路由



3.1 VLAN 基础配置及 Access 接口

原理概述

早期的局域网技术是基于总线型结构的。总线型拓扑结构是由一根单电缆连接着所有主机，这种局域网技术存在着冲突域问题，即所有用户都在一个冲突域中，那么同一时间内只有一台主机能发送消息，从任意设备发出的消息都会被其他所有主机接收到，用户可能收到大量不需要的报文；而且所有主机共享一条传输通道，任意主机之间都可以直接互相访问，无法控制信息的安全。

为了避免冲突域，同时扩展传统局域网以接入更多计算机，可以在局域网中使用二层交换机。交换机能有效隔离冲突域，但是由于所有计算机仍处于同一个广播域，任意设备都能接收到所有报文，不但降低了网络的效率，而且降低了安全性，即广播域和信息安全问题依旧存在。为了能减少广播，提高局域网安全性，人们使用虚拟局域网即 VLAN 技术把一个物理的 LAN 在逻辑上划分成多个广播域。VLAN 内的主机间可以直接通信，而 VLAN 间不能直接互通。这样，广播报文被限制在一个 VLAN 内，同时也提高了网络安全性。不同的 VLAN 使用不同的 VLAN ID 区分，VLAN ID 的范围是 0～4095，可配置的值为 1～4094，0 和 4095 为保留值。

Access 接口是交换机上用来连接用户主机的接口。当 Access 接口从主机收到一个不带 VLAN 标签的数据帧时，会给该数据帧加上与 PVID 一致的 VLAN 标签（PVID 可手工配置，默认是 1，即所有交换机上的接口默认都属于 VLAN 1）。当 Access 接口要发送一个带 VLAN 标签的数据帧给主机时，首先检查该数据帧的 VLAN ID 是否与自己的 PVID 相同，若相同，则去掉 VLAN 标签后发送该数据帧给主机；若不相同，直接丢弃该数据帧。

实验目的

- 理解 VLAN 的应用场景
- 掌握 VLAN 的基本配置
- 掌握 Access 接口的配置方法
- 掌握 Access 接口加入相应 VLAN 的方法

实验内容

本实验模拟企业网络场景。公司内网是一个大的局域网，二层交换机 S1 放置在一楼，在一楼办公的部门有 IT 部和人事部；二层交换机 S2 放置在二楼，在二楼办公的部门有市场部和研发部。由于交换机组成的是一个广播网，交换机连接的所有主机都能互相通信，而公司策略是：不同部门之间的主机不能互相通信，同一部门内的主机才可以互相访问。因此需要在交换机上划分不同的 VLAN，并将连接主机的交换机接口配置成 Access 接口划分到相应 VLAN 内。

实验拓扑

VLAN 基础配置及 Access 接口拓扑如图 3-1 所示。

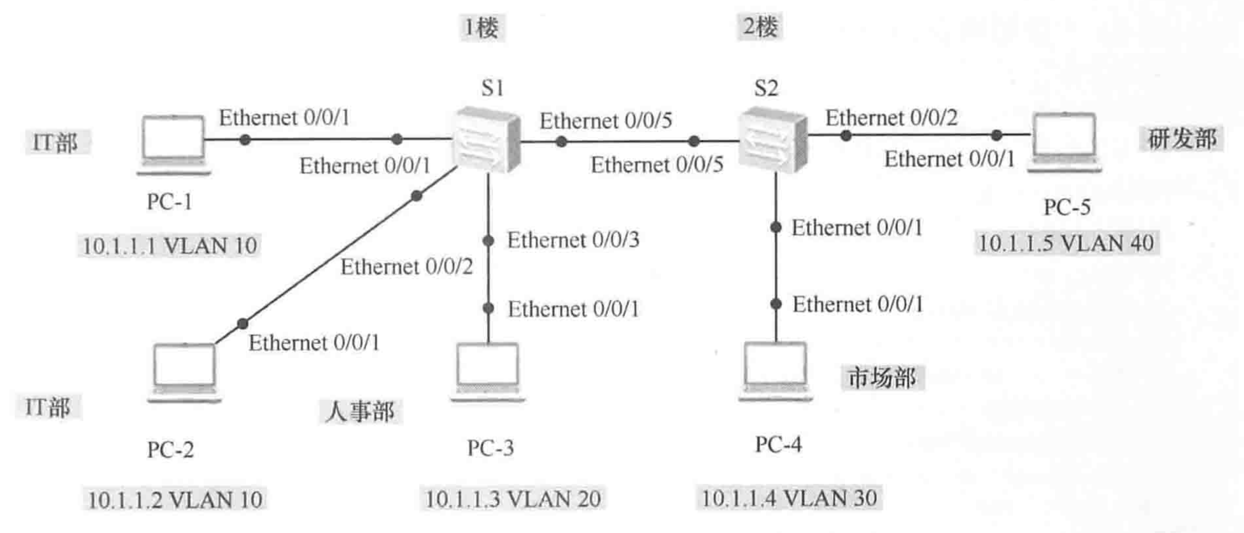


图 3-1　VLAN 基础配置及 Access 接口拓扑

实验编址

实验编址见表 3-1。

表 3-1　　实验编址

设备	接口	IP 地址	子网掩码	默认网关
PC-1	Ethernet 0/0/1	10.1.1.1	255.255.255.0	N/A
PC-2	Ethernet 0/0/1	10.1.1.2	255.255.255.0	N/A
PC-3	Ethernet 0/0/1	10.1.1.3	255.255.255.0	N/A
PC-4	Ethernet 0/0/1	10.1.1.4	255.255.255.0	N/A
PC-5	Ethernet 0/0/1	10.1.1.5	255.255.255.0	N/A

实验步骤

1. 基本配置

根据实验编址进行相应的基本 IP 地址配置，在此步骤中不要为交换机创建任何的 VLAN。

使用 **ping** 命令检测各直连链路的连通性，所有的 PC 都能相互通信。

```
[PC]ping -c 1 10.1.1.2
  PING 10.1.1.2: 56   data bytes, press CTRL_C to break
 Reply from 10.1.1.2: bytes=56 Sequence=1 ttl=255 time=50 ms
  --- 10.1.1.2 ping statistics ---
    1 packet(s) transmitted
    1 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 50/50/50 ms
```

其他主机间互相通信测试和上述相同，略过。

2. 创建 VLAN

除默认 VLAN 1 外，其余 VLAN 需要通过命令来手工创建。创建 VLAN 有两种方式，一种是使用 **vlan** 命令一次创建单个 VLAN，另一种方式是使用 **vlan batch** 命令一次创建多个 VLAN。

在 S1 上使用两条命令分别创建 VLAN 10 和 VLAN 20。

```
[S1]vlan 10
[S1-vlan10]vlan 20
```

在 S2 上使用一条 **vlan bath** 命令创建 VLAN 30 和 VLAN 40。

```
[S2]vlan batch 30 40
```

配置完成后，在 S1 和 S2 上使用 **display vlan** 命令查看 VLAN 的相关信息。

```
[S1]display vlan
The total number of vlans is : 3
--------------------------------------------------------------------------------
U: Up;          D: Down;          TG: Tagged;          UT: Untagged;
MP: Vlan-mapping;                 ST: Vlan-stacking;
#: ProtocolTransparent-vlan;      *: Management-vlan;
--------------------------------------------------------------------------------
VID  Type     Ports
--------------------------------------------------------------------------------
1    common  UT:Eth0/0/1(D)     Eth0/0/2(D)      Eth0/0/3(D)      Eth0/0/4(D)
                Eth0/0/5(D)      Eth0/0/6(D)      Eth0/0/7(D)      Eth0/0/8(D)
                Eth0/0/9(D)      Eth0/0/10(D)     Eth0/0/11(D)     Eth0/0/12(D)
                Eth0/0/13(D)     Eth0/0/14(D)     Eth0/0/15(D)     Eth0/0/16(D)
                Eth0/0/17(D)     Eth0/0/18(D)     Eth0/0/19(D)     Eth0/0/20(D)
                Eth0/0/21(D)     Eth0/0/22(D)     GE0/0/1(D)       GE0/0/2(D)
10   common
20   common

[S2]display vlan
The total number of vlans is : 3
--------------------------------------------------------------------------------
U: Up;          D: Down;          TG: Tagged;          UT: Untagged;
MP: Vlan-mapping;                 ST: Vlan-stacking;
#: ProtocolTransparent-vlan;      *: Management-vlan;
--------------------------------------------------------------------------------
VID  Type     Ports
--------------------------------------------------------------------------------
1    common  UT:Eth0/0/1(D)     Eth0/0/2(D)      Eth0/0/3(D)      Eth0/0/4(D)
                Eth0/0/5(D)      Eth0/0/6(D)      Eth0/0/7(D)      Eth0/0/8(D)
                Eth0/0/9(D)      Eth0/0/10(D)     Eth0/0/11(D)     Eth0/0/12(D)
                Eth0/0/13(D)     Eth0/0/14(D)     Eth0/0/15(D)     Eth0/0/16(D)
                Eth0/0/17(D)     Eth0/0/18(D)     Eth0/0/19(D)     Eth0/0/20(D)
                Eth0/0/21(D)     Eth0/0/22(D)     GE0/0/1(D)       GE0/0/2(D)
30   common
40   common
```

可以观察到，S1 和 S2 都已经成功创建了相应 VLAN，但目前没有任何接口加入所创建的 VLAN 10 与 20 中，默认情况下交换机上所有接口都属于 VLAN 1。

3. 配置 Access 接口

按照拓扑，使用 **port link-type access** 命令配置所有 S1 和 S2 交换机上连接 PC 的接口类型为 Access 类型接口，并使用 **port default vlan** 命令配置接口的默认 VLAN 并同时加入相应 VLAN 中。默认情况下，所有接口的默认 VLAN ID 为 1。

```
[S1]interface ethernet0/0/1
[S1-Ethernet0/0/1]port link-type access
[S1-Ethernet0/0/1]port default vlan 10
[S1-Ethernet0/0/1]interface ethernet0/0/2
[S1-Ethernet0/0/2]port link-type access
[S1-Ethernet0/0/2]port default vlan 10
[S1-Ethernet0/0/2]interface ethernet0/0/3
[S1-Ethernet0/0/3]port link-type access
[S1-Ethernet0/0/3]port default vlan 20

[S2]interface ethernet0/0/1
[S2-Ethernet0/0/1]port link-type access
[S2-Ethernet0/0/1]port default vlan 30
[S2-Ethernet0/0/1]interface ethernet0/0/2
[S2-Ethernet0/0/2]port link-type access
[S2-Ethernet0/0/2]port default vlan 40
```

配置完成后，查看 S1 与 S2 上的 VLAN 信息。

```
[S1]display vlan
The total number of vlans is : 3
--------------------------------------------------------------------------------
U: Up;          D: Down;          TG: Tagged;          UT: Untagged;
MP: Vlan-mapping;                 ST: Vlan-stacking;
#: ProtocolTransparent-vlan;    *: Management-vlan;
--------------------------------------------------------------------------------
VID  Type    Ports
1    common  UT:Eth0/0/4(D)     Eth0/0/5(D)     Eth0/0/6(D)     Eth0/0/7(D)
                Eth0/0/8(D)     Eth0/0/9(D)     Eth0/0/10(D)    Eth0/0/11(D)
                Eth0/0/12(D)    Eth0/0/13(D)    Eth0/0/14(D)    Eth0/0/15(D)
                Eth0/0/16(D)    Eth0/0/17(D)    Eth0/0/18(D)    Eth0/0/19(D)
                Eth0/0/20(D)    Eth0/0/21(D)    Eth0/0/22(D)    GE0/0/1(D)
                GE0/0/2(D)
10   common  UT:Eth0/0/1(D)     Eth0/0/2(D)
20   common  UT:Eth0/0/3(D)

[S2]display vlan
The total number of vlans is : 3
--------------------------------------------------------------------------------
U: Up;          D: Down;          TG: Tagged;          UT: Untagged;
MP: Vlan-mapping;                 ST: Vlan-stacking;
#: ProtocolTransparent-vlan;    *: Management-vlan;
--------------------------------------------------------------------------------
VID  Type    Ports
--------------------------------------------------------------------------------
1    common  UT: Eth0/0/3(D)    Eth0/0/4(D)     Eth0/0/5(D)     Eth0/0/6(D)
                 Eth0/0/7(D)    Eth0/0/8(D)     Eth0/0/9(D)     Eth0/0/10(D)
                 Eth0/0/11(D)   Eth0/0/12(D)    Eth0/0/13(D)    Eth0/0/14(D)
                 Eth0/0/15(D)   Eth0/0/16(D)    Eth0/0/17(D)    Eth0/0/18(D)
                 Eth0/0/19(D)   Eth0/0/20(D)    Eth0/0/21(D)    Eth0/0/22(D)
                 GE0/0/1(D)     GE0/0/2(D)
30   common  UT:Eth0/0/1(D)
40   common  UT:Eth0/0/2(D)
```

可以观察到，目前两台交换机上连接PC的接口都已经加入到相应所属部门的VLAN当中。

4. 检查配置结果

在交换机上将不同接口加入各自不同的 VLAN 中后，属于相同 VLAN 的接口处于

同一个广播域，相互之间可以直接通信。属于不同 VLAN 的接口是处于不同的广播域，相互之间不能直接通信。

在本实验环境中，只有同属于 IT 部门 VLAN 10 的两台主机 PC-1 和 PC-2 之间可以互相通信。其他不同部门间的 PC 之间将无法通信。

在 IT 部门的终端 PC-1 上分别测试与同部门的终端 PC-2、HR 部门的 PC-3 间的连通性。

```
PC>ping -c 1 10.1.1.2
  PING 10.0.1.2: 56    data bytes, press CTRL_C to break
 Reply from 10.1.1.2: bytes=56 Sequence=1 ttl=255 time=50 ms
  --- 10.1.1.2 ping statistics ---
    1 packet(s) transmitted
    1 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 50/50/50 ms

PC>ping 10.1.1.3
From 0.0.0.0: Destination host unreachable
From 0.0.0.0: Destination host unreachable
From 0.0.0.0: Destination host unreachable
From 0.0.0.0: Destination host unreachable
From 0.0.0.0: Destination host unreachable
……
```

可以观察到，相同 VLAN 内的 PC 可以互相通信，不同 VLAN 内的 PC 间无法通信。

思考

在本实验中，如果将 S2 的接口 E 0/0/5 配置为 Access 类型接口，并划入 VLAN 30 中，此时 PC-1 能否 ping 通 PC-4？PC-1 能否 ping 通 PC-5？为什么？

3.2 配置 Trunk 接口

原理概述

在以太网中，通过划分 VLAN 来隔离广播域和增强网络通信的安全性。以太网通常由多台交换机组成，为了使 VLAN 的数据帧跨越多台交换机传递，交换机之间互连的链路需要配置为干道链路（Trunk Link）。和接入链路不同，干道链路是用来在不同的设备之间（如交换机和路由器之间、交换机和交换机之间）承载多个不同 VLAN 数据的，它不属于任何一个具体的 VLAN，可以承载所有的 VLAN 数据，也可以配置为只能传输指定 VLAN 的数据。

Trunk 端口一般用于交换机之间连接的端口，Trunk 端口可以属于多个 VLAN，可以接收和发送多个 VLAN 的报文。

当 Trunk 端口收到数据帧时，如果该帧不包含 802.1Q 的 VLAN 标签，将打上该 Trunk 端口的 PVID；如果该帧包含 802.1Q 的 VLAN 标签，则不改变。

当 Trunk 端口发送数据帧时，当该所发送帧的 VLAN ID 与端口的 PVID 不同时，检查是否允许该 VLAN 通过，若允许的话直接透传，不允许就直接丢弃；当该帧的 VLAN ID 与端口的 PVID 相同时，则剥离 VLAN 标签后转发。

实验目的

- 理解干道链路的应用场景
- 掌握 Trunk 端口的配置
- 掌握 Trunk 端口允许所有 VLAN 通过的配置方法
- 掌握 Trunk 端口允许特定 VLAN 通过的配置方法

实验内容

本实验模拟某公司网络场景。公司规模较大，员工 200 余名，内部网络是一个大的局域网。公司放置了多台接入交换机（如 S1 和 S2）负责员工的网络接入。接入交换机之间通过汇聚交换机 S3 相连。公司通过划分 VLAN 来隔离广播域，由于员工较多，相同部门的员工通过不同交换机接入。为了保证在不同交换机下相同部门的员工能互相通信，需要配置交换机之间链路为干道模式，以实现相同 VLAN 跨交换机通信。

实验拓扑

跨交换机实现 VLAN 间通信的拓扑如图 3-2 所示。

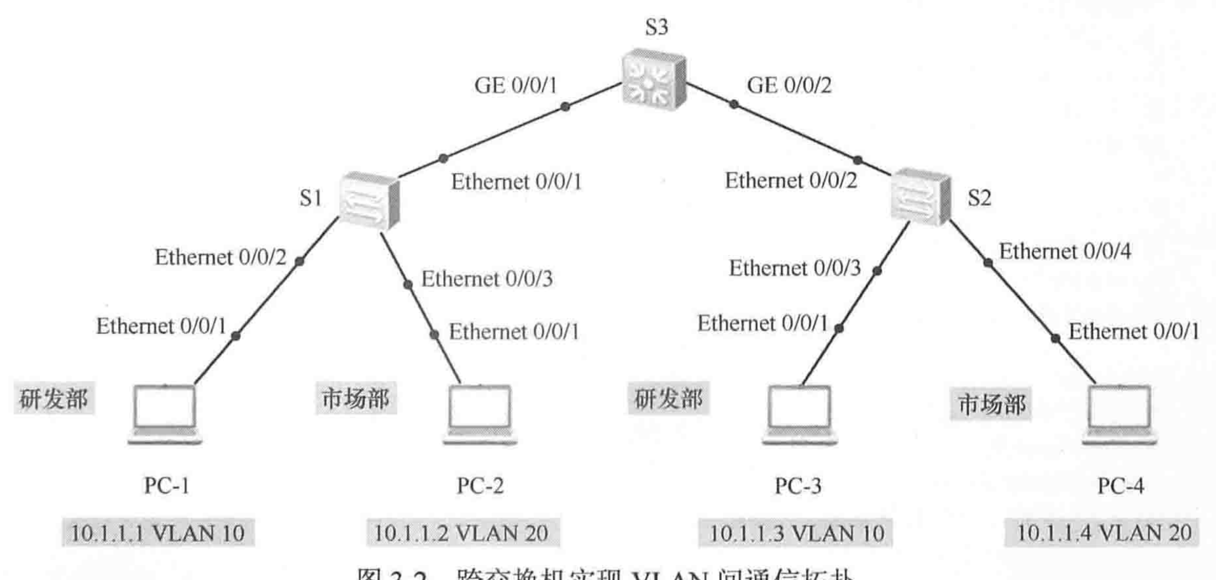


图 3-2 跨交换机实现 VLAN 间通信拓扑

实验编址

实验编址见表 3-2。

表 3-2 实验编址

设备	接口	IP 地址	子网掩码	默认网关
PC-1	Ethernet 0/0/1	10.1.1.1	255.255.255.0	N/A
PC-2	Ethernet 0/0/1	10.1.1.2	255.255.255.0	N/A
PC-3	Ethernet 0/0/1	10.1.1.3	255.255.255.0	N/A
PC-4	Ethernet 0/0/1	10.1.1.4	255.255.255.0	N/A

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 地址配置，并使用 **ping** 命令检测各直连链路的连通性。在没有完成划分 VLAN 之前各 PC 之间都能互通（属于默认 VLAN 1）。

这里以 PC-1 与 PC-3 的 ping 测试为例，其余省略。

```
PC>ping 10.1.1.3
Ping 10.1.1.3: 32 data bytes, Press Ctrl_C to break
From 10.1.1.3: bytes=32 seq=1 ttl=128 time=62 ms
From 10.1.1.3: bytes=32 seq=2 ttl=128 time=62 ms
From 10.1.1.3: bytes=32 seq=3 ttl=128 time=62 ms
From 10.1.1.3: bytes=32 seq=4 ttl=128 time=78 ms
From 10.1.1.3: bytes=32 seq=5 ttl=128 time=78 ms
--- 10.1.1.3 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 30/68/120 ms
```

2. 创建 VLAN，配置 Access 接口

公司内网需要通过 VLAN 的划分来隔离不同的部门，需要在 3 台交换机 S1、S2、S3 上都分别创建 VLAN 10 和 VLAN 20，研发部员工属于 VLAN 10，市场部员工属于 VLAN 20。

```
[S1]vlan 10
[S1-vlan10]description R$D
[S1-vlan10]vlan 20
[S1-vlan20]description Market

[S2]vlan 10
[S2-vlan10]description R$D
[S2-vlan10]vlan 20
[S2-vlan20]description Market

[S3]vlan 10
[S3-vlan10]description R$D
[S3-vlan10]vlan 20
[S3-vlan20]description Market
```

配置完成后，使用 **display vlan** 命令查看所配置的 VLAN 信息，以 S3 为例。

```
<S3>display vlan
The total number of vlans is : 3
--------------------------------------------------------------------------------
U: Up;          D: Down;          TG: Tagged;          UT: Untagged;
MP: Vlan-mapping;                 ST: Vlan-stacking;
#: ProtocolTransparent-vlan;      *: Management-vlan;
--------------------------------------------------------------------------------
VID  Type    Ports
--------------------------------------------------------------------------------
1    common  UT:GE0/0/1(U)     GE0/0/2(U)     GE0/0/3(D)     GE0/0/4(D)
                GE0/0/5(D)     GE0/0/6(D)     GE0/0/7(D)     GE0/0/8(D)
                GE0/0/9(D)     GE0/0/10(D)    GE0/0/11(D)    GE0/0/12(D)
                GE0/0/13(D)    GE0/0/14(D)    GE0/0/15(D)    GE0/0/16(D)
                GE0/0/17(D)    GE0/0/18(D)    GE0/0/19(D)    GE0/0/20(D)
                GE0/0/21(D)    GE0/0/22(D)    GE0/0/23(D)    GE0/0/24(D)
```

```
10    common
20    common
VID  Status  Property         MAC-LRN  Statistics  Description
1    enable  default          enable   disable     VLAN 0001
10   enable  default          enable   disable     R$D
20   enable  default          enable   disable     Market
```

可以观察到相关的 VLAN 都已经配置好。也可以使用 **display vlan summary** 命令查看所配置 VLAN 的简要信息。

```
<S3>display vlan summary
static vlan:
Total 3 static vlan.
  1 10 20
dynamic vlan:
Total 0 dynamic vlan.
reserved vlan:
Total 0 reserved vlan.
```

在 S1 上配置 E 0/0/2 和 E 0/0/3 为 Access 接口，并划分到相应的 VLAN。

```
[S1]interface Ethernet0/0/2
[S1-Ethernet0/0/2]port link-type access
[S1-Ethernet0/0/2]port default vlan 10

[S1]interface Ethernet0/0/3
[S1-Ethernet0/0/3]port link-type access
[S1-Ethernet0/0/3]port default vlan 20
```

在 S2 上配置 E 0/0/3 和 E 0/0/4 为 Access 接口，并划分到相应的 VLAN。

```
[S2]interface Ethernet0/0/3
[S2-Ethernet0/0/3]port link-type access
[S2-Ethernet0/0/3]port default vlan 10

[S2]interface Ethernet0/0/4
[S2-Ethernet0/0/4]port link-type access
[S2-Ethernet0/0/4]port default vlan 20
```

配置完成后，使用 **display port vlan** 命令检查 VLAN 和接口配置情况。

```
[S1]display port vlan
Port                Link Type    PVID  Trunk VLAN List
Ethernet0/0/1       hybrid       1     -
Ethernet0/0/2       access       10    -
Ethernet0/0/3       access       20    -

[S2]display port vlan
Port                Link Type    PVID  Trunk VLAN List
Ethernet0/0/1       hybrid       1     -
Ethernet0/0/2       hybrid       1     -
Ethernet0/0/3       access       10    -
Ethernet0/0/4       access       20    -
```

可以观察到 PC 所连接的交换机接口都已经被配置成 Access 模式，并且已经加入到了正确的 VLAN 中。

3. 配置 Trunk 接口

将 PC 所连入的交换机接口划入到相应的部门 VLAN 后，测试相同部门中的 PC 是否能够通信。

测试 PC-1 与 PC-3 之间的连通性。

```
PC>ping 10.1.1.3
Ping 10.1.1.3: 32 data bytes, Press Ctrl_C to break
From 10.1.1.1: Destination host unreachable
From 10.1.1.1: Destination host unreachable
From 10.1.1.1: Destination host unreachable
From 10.1.1.1: Destination host unreachable
From 10.1.1.1: Destination host unreachable
......
```

测试 PC-2 与 PC-4 之间的连通性。

```
PC>ping 10.1.1.4
Ping 10.1.1.4: 32 data bytes, Press Ctrl_C to break
From 10.1.1.2: Destination host unreachable
From 10.1.1.2: Destination host unreachable
From 10.1.1.2: Destination host unreachable
From 10.1.1.2: Destination host unreachable
From 10.1.1.2: Destination host unreachable
......
```

可以观察到此时同部门的 PC 间不能通信。

目前在该跨交换机实现不同 VLAN 通信的二层组网拓扑中，虽然与 PC 端相连的交换机接口上创建并划分了 VLAN 信息，但是在交换机与交换机之间相连的接口上并没有相应的 VLAN 信息，不能够识别和发送跨越交换机的 VLAN 报文，此时 VLAN 只具有在每台交换机上的本地意义，无法实现相同 VLAN 的跨交换机通信。

为了让交换机间能够识别和发送跨越交换机的 VLAN 报文，需要将交换机间相连的接口配置成为 Trunk 接口。配置时要明确被允许通过的 VLAN，实现对 VLAN 流量传输的控制。

在 S1 上配置 E 0/0/1 为 Trunk 接口，允许 VLAN 10 和 VLAN 20 通过。

```
[S1]interface Ethernet0/0/1
[S1-Ethernet0/0/1]port link-type trunk
[S1-Ethernet0/0/1]port trunk allow-pass vlan 10 20
```

在 S2 上配置 E 0/0/2 为 Trunk 接口，允许 VLAN 10 和 VLAN 20 通过。

```
[S2]interface Ethernet0/0/2
[S2-Ethernet0/0/2]port link-type trunk
[S2-Ethernet0/0/2]port trunk allow-pass vlan 10 20
```

在 S3 上配置 GE 0/0/1 和 GE 0/0/2 为 Trunk 接口，允许所有 VLAN 通过。

```
[S3]interface GigabitEthernet0/0/1
[S3-GigabitEthernet0/0/1]port link-type trunk
[S3-GigabitEthernet0/0/1]port trunk allow-pass vlan all

[S3]interface GigabitEthernet0/0/2
[S3-GigabitEthernet0/0/2]port link-type trunk
[S3-GigabitEthernet0/0/2]port trunk allow-pass vlan all
```

配置完成后可以使用 **display port vlan** 命令来检查 Trunk 的配置情况，这里以 S3 为例。

```
[S3]display port vlan
Port                    Link Type    PVID   Trunk VLAN List
GigabitEthernet0/0/1    trunk        1      1-4094
GigabitEthernet0/0/2    trunk        1      1-4094
```

可以观察到 S3 的 GE 0/0/1 和 GE0/0/2 已被成功配置为 Trunk 接口，并且允许所有 VLAN 流量通过（VLAN 1～4094）。

再次验证不同交换机上的相同部门的 PC 间的连通性。

测试 PC-1 与 PC-3 之间的连通性。

```
PC>ping 10.1.1.3
Ping 10.1.1.3: 32 data bytes, Press Ctrl_C to break
From 10.1.1.3: bytes=32 seq=1 ttl=128 time=46 ms
From 10.1.1.3: bytes=32 seq=2 ttl=128 time=78 ms
From 10.1.1.3: bytes=32 seq=3 ttl=128 time=78 ms
From 10.1.1.3: bytes=32 seq=4 ttl=128 time=46 ms
From 10.1.1.3: bytes=32 seq=5 ttl=128 time=47 ms
-- 10.1.1.3 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 30/68/120 ms
```

测试 PC-2 与 PC-4 之间的连通性。

```
PC>ping 10.1.1.4
Ping 10.1.1.4: 32 data bytes, Press Ctrl_C to break
From 10.1.1.4: bytes=32 seq=1 ttl=128 time=46 ms
From 10.1.1.4: bytes=32 seq=2 ttl=128 time=47 ms
From 10.1.1.4: bytes=32 seq=3 ttl=128 time=46 ms
From 10.1.1.4: bytes=32 seq=4 ttl=128 time=78 ms
From 10.1.1.4: bytes=32 seq=5 ttl=128 time=63 ms
-- 10.1.1.4 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 30/68/120 ms
```

可以观察到此时同部门中的 PC 已经能成功通信。

思考

连接 PC 的交换机接口也可以配置成 Trunk 接口吗？为什么？

3.3 理解 Hybrid 接口的应用

原理概述

Hybrid 接口既可以连接普通终端的接入链路又可以连接交换机间的干道链路，它允许多个 VLAN 的帧通过，并可以在出接口方向将某些 VLAN 帧的标签剥掉。

Hybrid 接口处理 VLAN 帧的过程如下：

（1）收到一个二层帧，判断是否有 VLAN 标签。没有标签，则标记上 Hybrid 接口的 PVID，进行下一步处理；有标签，判断该 Hybrid 接口是否允许该 VLAN 的帧进入，允许则进行下一步处理，否则丢弃。

（2）当数据帧从 Hybrid 接口发出时，交换机判断 VLAN 在本接口的属性是 Untagged 还是 Tagged。如果是 Untagged，先剥离帧的 VLAN 标签，再发送；如果是 Tagged，则直接发送帧。

通过配置 Hybrid 接口，能够实现对 VLAN 标签的灵活控制，既能够实现 Access 接口的功能，又能够实现 Trunk 接口的功能。

实验目的

- 掌握配置 Hybrid 接口的方法
- 理解 Hybrid 接口处理 Untagged 数据帧过程
- 理解 Hybrid 接口处理 Tagged 数据帧过程
- 理解 Hybrid 接口的应用场景

实验内容

某企业二层网络使用两台 S3700 交换机 S1 和 S2，且两台设备在不同的楼层。网络管理员规划了 3 个不同 VLAN，HR 部门使用 VLAN 10，市场部门使用 VLAN 20，IT 部门使用 VLAN 30。现在需要让处于不同楼层的 HR 部门和市场部门实现部门内部通信，而两部门之间不允许互相通信；IT 部门可以访问任意部门。可以通过配置 Hybrid 接口来实现较复杂的 VLAN 控制。

实验拓扑

理解 Hybrid 接口的应用拓扑如图 3-3 所示。

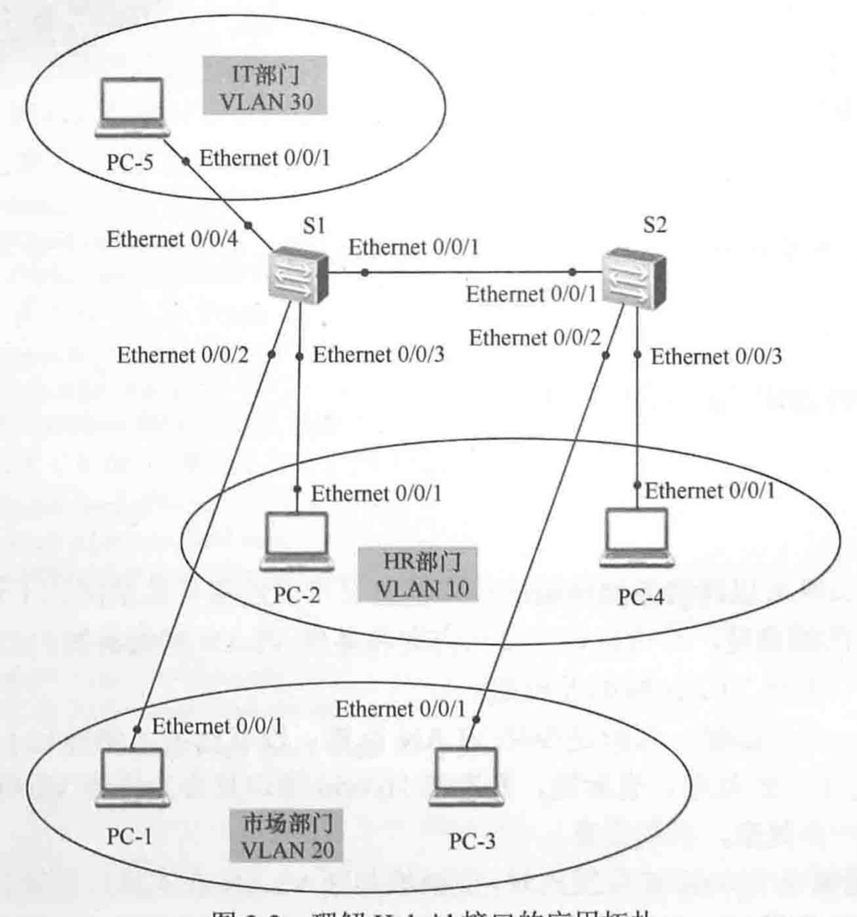


图 3-3 理解 Hybrid 接口的应用拓扑

实验编址

实验编址见表 3-3。

表 3-3 实验编址

设备	接口	IP 地址	子网掩码	默认网关
PC-1	Ethernet 0/0/1	192.168.1.1	255.255.255.0	N/A
PC-2	Ethernet 0/0/1	192.168.1.2	255.255.255.0	N/A
PC-3	Ethernet 0/0/1	192.168.1.3	255.255.255.0	N/A
PC-4	Ethernet 0/0/1	192.168.1.4	255.255.255.0	N/A
PC-5	Ethernet 0/0/1	192.168.1.100	255.255.255.0	N/A

实验步骤

1. 基本配置

按照实验编址表为 PC 配置 IP 地址，如图 3-4 所示的配置过程适用于所有终端。

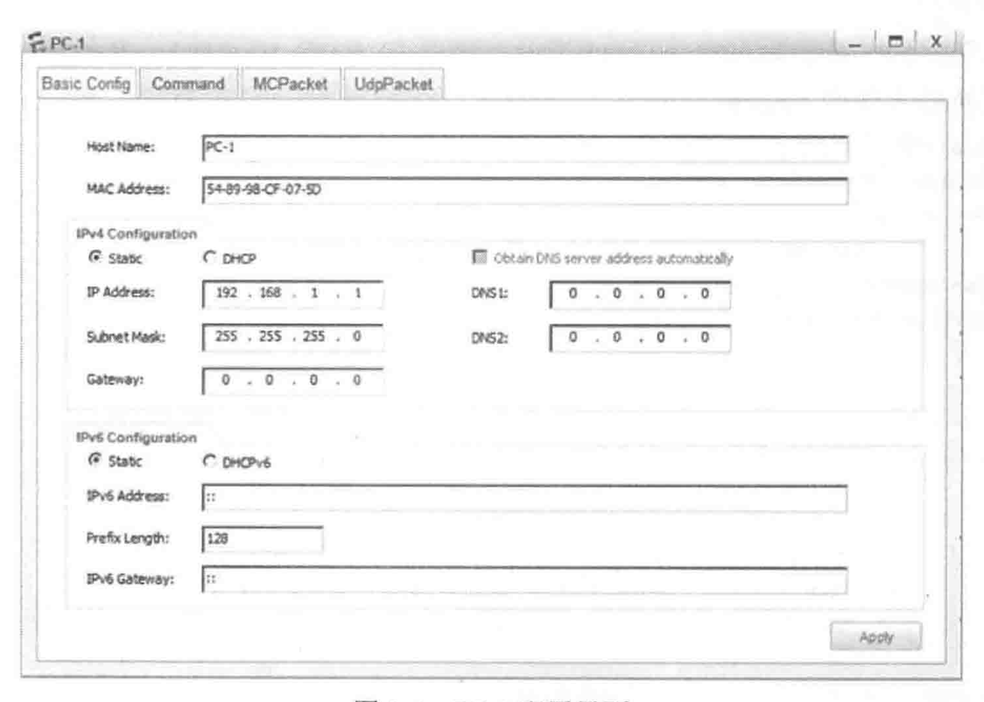


图 3-4 PC-1 配置界面

完成配置后，测试主机之间的连通性。

在 PC-1 上，使用 **ping** 命令。

```
PC>ping 192.168.1.2
Ping 192.168.1.2: 32 data bytes, Press Ctrl_C to break
From 192.168.1.2: bytes=32 seq=1 ttl=128 time=15 ms
From 192.168.1.2: bytes=32 seq=2 ttl=128 time<1 ms
From 192.168.1.2: bytes=32 seq=3 ttl=128 time<1 ms
From 192.168.1.2: bytes=32 seq=4 ttl=128 time=15 ms
From 192.168.1.2: bytes=32 seq=5 ttl=128 time<1 ms
--- 192.168.1.2 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 0/6/15 ms
```

可以观察到，此时 PC-1 访问其他主机通信正常，其他主机上的测试过程省略。

在没有定义 VLAN 及接口类型之前，默认情况下，交换机上所有接口都是 Hybrid

类型，接口的 PVID 是 VLAN 1，即所有接口收到没有标签的二层数据帧，都被转发到 VLAN 1 中，并继续以 Untagged 的方式把帧发送至同为 VLAN 1 的其他接口。所以，即使未做任何配置，主机之间默认仍然可以互相通信。

在 S1 上使用 **display port vlan** 命令查看接口的默认类型。

```
<S1>display port vlan
Port                    Link Type    PVID  Trunk VLAN List
-------------------------------------------------------------------------------
Ethernet0/0/1           hybrid       1     -
Ethernet0/0/2           hybrid       1     -
Ethernet0/0/3           hybrid       1     -
Ethernet0/0/4           hybrid       1     -
Ethernet0/0/5           hybrid       1     -
Ethernet0/0/6           hybrid       1     -
……
```

可以观察到，接口默认是 Hybrid 类型，接口 PVID 是 VLAN 1，其他接口也一样。

在交换机上使用 **display vlan** 命令查看接口和所属 VLAN 的对应关系。

```
[S1]display vlan
The total number of vlans is : 1
-------------------------------------------------------------------------------
U: Up;          D: Down;          TG: Tagged;          UT: Untagged;
MP: Vlan-mapping;                 ST: Vlan-stacking;
#: ProtocolTransparent-vlan;    *: Management-vlan;
-------------------------------------------------------------------------------
VID  Type     Ports
-------------------------------------------------------------------------------
1    common   UT:Eth0/0/1(U)     Eth0/0/2(U)     Eth0/0/3(U)     Eth0/0/4(U)
                 Eth0/0/5(D)     Eth0/0/6(D)     Eth0/0/7(D)     Eth0/0/8(D)
                 Eth0/0/9(D)     Eth0/0/10(D)    Eth0/0/11(D)    Eth0/0/12(D)
                 Eth0/0/13(D)    Eth0/0/14(D)    Eth0/0/15(D)    Eth0/0/16(D)
                 Eth0/0/17(D)    Eth0/0/18(D)    Eth0/0/19(D)    Eth0/0/20(D)
                 Eth0/0/21(D)    Eth0/0/22(D)    GE0/0/1(D)      GE0/0/2(D)

VID  Status  Property      MAC-LRN Statistics Description
1    enable  default       enable  disable    VLAN 0001
```

可以观察到，所有接口都默认属于 VLAN 1，其他交换机也都一样，因此 VLAN 1 内所有的主机都可以直接访问。

2. 实现组内通信、组间隔离

交换机接口的类型可以是 Access、Trunk 和 Hybrid。Access 类型的接口仅属于一个 VLAN，只能接收、转发相应 VLAN 的帧；而 Trunk 类型接口则默认属于所有 VLAN，任何 Tagged 帧都能经过 Trunk 接收和转发；Hybrid 类型接口则介于二者之间，可自主定义端口上能接收和转发哪些 VLAN Tag 的帧，并可决定 VLAN Tag 是否继续携带或者剥离。Access 和 Trunk 类型接口是 Hybrid 类型接口的两个特例，一个仅支持一个 VLAN 的传递，一个默认支持所有 VLAN 的传递，而 Access 类型和 Trunk 类型的接口能做到的，Hybrid 接口都能做到。

目前要求实现 HR 部门和市场部门的员工终端可以进行部门内部通信，即 VLAN 10 内 PC-2 和 PC-4 之间可以自由访问，VLAN 20 内 PC-1 和 PC-3 之间可以自由访问，而两个部门间的员工不能互相访问，即 VLAN 10 和 VLAN 20 之间不能互相访问。要实现此

需求，可以使用 Access 和 Trunk 的配置方法，也可以仅使用 Hybrid 的配置方法。

使用 Trunk 和 Access 类型接口的配置过程如下。

将 S1 上的 E 0/0/2 和 S2 上的 E 0/0/2 配置为 Access 类型，并将相应的接口加入到 VLAN 20。同理，将 S1 上的 E 0/0/3 和 S2 上的 E 0/0/3 也配置为 Access 类型接口，并加入到 VLAN 10。而交换机之间的互连链路的两个 E 0/0/1 接口则配置为 Trunk 类型。

```
[S1]vlan 10
[S1-vlan10]vlan 20
[S1-vlan20]interface Ethernet0/0/3
[S1-Ethernet0/0/3]port link-type access
[S1-Ethernet0/0/3]port default vlan 10
[S1-Ethernet0/0/3]interface ethernet0/0/2
[S1-Ethernet0/0/2]port link-type access
[S1-Ethernet0/0/2]port default vlan 20
[S1-Ethernet0/0/2]interface Ethernet0/0/1
[S1-Ethernet0/0/1]port link-type trunk
[S1-Ethernet0/0/1]port trunk allow-pass vlan all

[S2]vlan 10
[S2-vlan10]vlan 20
[S2-vlan10]interface Ethernet0/0/3
[S2-Ethernet0/0/3]port link-type access
[S2-Ethernet0/0/3]port default vlan 10
[S2-Ethernet0/0/3]interface ethernet0/0/2
[S2-Ethernet0/0/2]port link-type access
[S2-Ethernet0/0/2]port default vlan 20
[S2-Ethernet0/0/2]interface ethernet0/0/1
[S2-Ethernet0/0/1]port link-type trunk
[S2-Ethernet0/0/1]port trunk allow-pass vlan all
```

配置完成后，查看接口和 VLAN 的对应关系。

```
[S1]display vlan
The total number of vlans is : 3
--------------------------------------------------------------------------------
U: Up;          D: Down;           TG: Tagged;          UT: Untagged;
MP: Vlan-mapping;                  ST: Vlan-stacking;
#: ProtocolTransparent-vlan;    *: Management-vlan;
--------------------------------------------------------------------------------
VID  Type     Ports
--------------------------------------------------------------------------------
1    common   UT:Eth0/0/1(U)     Eth0/0/4(U)     Eth0/0/5(D)     Eth0/0/6(D)
                 Eth0/0/7(D)     Eth0/0/8(D)     Eth0/0/9(D)     Eth0/0/10(D)
                 Eth0/0/11(D)    Eth0/0/12(D)    Eth0/0/13(D)    Eth0/0/14(D)
                 Eth0/0/15(D)    Eth0/0/16(D)    Eth0/0/17(D)    Eth0/0/18(D)
                 Eth0/0/19(D)    Eth0/0/20(D)    Eth0/0/21(D)    Eth0/0/22(D)
                 GE0/0/1(D)      GE0/0/2(D)
10   common   UT:Eth0/0/3(U)
20   common   UT:Eth0/0/2(U)
VID  Status  Property      MAC-LRN Statistics Description
1    enable  default       enable  disable    VLAN 0001
10   enable  default       enable  disable    VLAN 0010
20   enable  default       enable  disable    VLAN 0020

[S2]display vlan
```

```
The total number of vlans is : 3
--------------------------------------------------------------------------------
U: Up;          D: Down;          TG: Tagged;          UT: Untagged;
MP: Vlan-mapping;                 ST: Vlan-stacking;
#: ProtocolTransparent-vlan;      *: Management-vlan;
--------------------------------------------------------------------------------
VID  Type    Ports
--------------------------------------------------------------------------------
1    common  UT:Eth0/0/1(U)     Eth0/0/4(D)     Eth0/0/5(D)     Eth0/0/6(D)
                Eth0/0/7(D)     Eth0/0/8(D)     Eth0/0/9(D)     Eth0/0/10(D)
                Eth0/0/11(D)    Eth0/0/12(D)    Eth0/0/13(D)    Eth0/0/14(D)
                Eth0/0/15(D)    Eth0/0/16(D)    Eth0/0/17(D)    Eth0/0/18(D)
                Eth0/0/19(D)    Eth0/0/20(D)    Eth0/0/21(D)    Eth0/0/22(D)
                GE0/0/1(D)      GE0/0/2(D)
10   common  UT:Eth0/0/3(U)
20   common  UT:Eth0/0/2(U)
VID  Status  Property     MAC-LRN Statistics Description
1    enable  default      enable  disable    VLAN 0001
10   enable  default      enable  disable    VLAN 0010
20   enable  default      enable  disable    VLAN 0020
```

可以观察到，配置已经生效。

在 PC-1 上测试与同 VLAN 20 的 PC-3 的连通性，以及与 VLAN 10 内终端的连通性。

```
PC>ping 192.168.1.3
Ping 192.168.1.3: 32 data bytes, Press Ctrl_C to break
From 192.168.1.3: bytes=32 seq=1 ttl=128 time=47 ms
From 192.168.1.3: bytes=32 seq=2 ttl=128 time=31 ms
From 192.168.1.3: bytes=32 seq=3 ttl=128 time=46 ms
From 192.168.1.3: bytes=32 seq=4 ttl=128 time=31 ms
From 192.168.1.3: bytes=32 seq=5 ttl=128 time=31 ms
--- 192.168.1.3 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 31/37/47 ms

PC>ping 192.168.1.4
Ping 192.168.1.4: 32 data bytes, Press Ctrl_C to break
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
......
```

可以观察到，在单台交换机及跨交换机间的访问控制使用 Trunk 和 Access 类型接口实现了需求，但同样的需求使用 Hybrid 实现会更灵活。

S1 的 E 0/0/2 接口连接 PC-1 主机，该接口收到的 PC-1 发送的 Untagged 的帧会被交换机转发到 VLAN 20。同样，交换机从其他接口收到 VLAN 20 的发往 PC-1 的帧也会以 Untagged 的方式从 E 0/0/2 接口发送。S1 的 E 0/0/3 接口连接 PC-2 主机，该接口收到 Untagged 的帧会被转发到 VLAN 10。如果交换机收到的 VLAN 10 的发往 PC-2 的帧也会以 Untagged 的方式从接口 E 0/0/3 发送。VLAN 10 和 VLAN 20 的帧也要经过交换机间链路发送至邻居交换机 S2。反之，S1 收到来自邻居交换机 S2 的 Tagged 的帧后，也会

根据 VLAN Tag 转发到相应的 VLAN。

在 S1 的 E 0/0/2 接口上使用 **undo port default vlan** 命令用来恢复接口默认 VLAN。

```
[S1]interface ethernet0/0/2
[S1-Ethernet0/0/2]undo port default vlan
```

配置 **port link-type hybrid** 命令修改接口类型为默认的 Hybrid 类型。

```
[S1-Ethernet0/0/2]port link-type hybrid
```

配置 **port hybrid untagged vlan 20** 命令使得交换机在该接口转发 VLAN 20 的帧时，剥离掉相应的 VLAN Tag 20，以 Untagged 的方式发送给 PC。

```
[S1-Ethernet0/0/2]port hybrid untagged vlan 20
```

配置 **port hybrid pvid vlan 20** 命令设置 Hybrid 类型接口的默认 VLAN ID，即使得该端口上接收到 PC 发来的未带 VLAN Tag 的帧时，加上 VLAN Tag 20，并转发到 VLAN 20。

```
[S1-Ethernet0/0/2]port hybrid pvid vlan 20
```

同样在连接另一台终端的 E 0/0/3 接口做同样配置。

```
[S1]interface ethernet0/0/3
[S1-Ethernet0/0/3]undo port default vlan
[S1-Ethernet0/0/3]port link-type hybrid
[S1-Ethernet0/0/3]port hybrid untagged vlan 10
[S1-Ethernet0/0/3]port hybrid pvid vlan 10
```

在连接交换机 S2 的 E 0/0/1 接口上修改端口类型为默认的 Hybrid 类型，并使用 **port hybrid tagged vlan 10 20** 命令设置该链路仅接收带有 VLAN Tag 10 和 20 的帧，而交换机也仅转发 VLAN 10 和 VLAN 20 的帧到该链路。一般该命令配置在交换机互连的链路接口之上。

```
[S1]interface ethernet0/0/1
[S1-Ethernet0/0/1]port trunk allow-pass vlan 1
[S1-Ethernet0/0/1]port link-type hybrid
[S1-Ethernet0/0/1]port hybrid tagged vlan 10 20
```

S2 交换机将在 E 0/0/1 接口接收到的 Tagged 帧，根据 VLAN Tag 标识，向接口 E 0/0/2 转发 VLAN 20 的帧，向接口 E 0/0/3 转发 VLAN 30 的帧。反之，接口 E 0/0/2 接收到 PC 发送的未带 Tag 的帧转发到 VLAN 20，端口 E 0/0/3 接收的到未带 Tag 的帧会被转发到 VLAN 10，并且这些帧发送到邻居交换机 S1 时，会保留原有 Tag。

S2 上的配置和 S1 类似，这里不再解释。

```
[S2]interface ethernet0/0/2
[S2-Ethernet0/0/2]undo port default vlan
[S2-Ethernet0/0/2]port link-type hybrid
[S2-Ethernet0/0/2]port hybrid untagged vlan 20
[S2-Ethernet0/0/2]port hybrid pvid vlan 20
[S2-Ethernet0/0/2]interface ethernet0/0/3
[S2-Ethernet0/0/3]undo port default vlan
[S2-Ethernet0/0/3]port link-type hybrid
[S2-Ethernet0/0/3]port hybrid untagged vlan 10
[S2-Ethernet0/0/3]port hybrid pvid vlan 10
[S2-Ethernet0/0/3]interface ethernet 0/0/1
[S2-Ethernet0/0/1]port trunk allow-pass vlan 1
[S2-Ethernet0/0/1]port link-type hybrid
[S2-Ethernet0/0/1]port hybrid tagged vlan 10 20
```

配置完成后，使用 **display vlan** 命令查看使用 Hybrid 配置下接口和 VLAN 的对应关系。

```
[S1]display vlan
The total number of vlans is : 3
--------------------------------------------------------------------------------
U: Up;          D: Down;          TG: Tagged;          UT: Untagged;
MP: Vlan-mapping;                 ST: Vlan-stacking;
#: ProtocolTransparent-vlan;      *: Management-vlan;
--------------------------------------------------------------------------------
VID  Type    Ports
--------------------------------------------------------------------------------
1    common  UT:Eth0/0/1(U)    Eth0/0/2(U)     Eth0/0/3(U)     Eth0/0/4(U)
                Eth0/0/5(D)    Eth0/0/6(D)     Eth0/0/7(D)     Eth0/0/8(D)
                Eth0/0/9(D)    Eth0/0/10(D)    Eth0/0/11(D)    Eth0/0/12(D)
                Eth0/0/13(D)   Eth0/0/14(D)    Eth0/0/15(D)    Eth0/0/16(D)
                Eth0/0/17(D)   Eth0/0/18(D)    Eth0/0/19(D)    Eth0/0/20(D)
                Eth0/0/21(D)   Eth0/0/22(D)    GE0/0/1(D)      GE0/0/2(D)
10   common  UT:Eth0/0/3(U)
 TG:Eth0/0/1(U)
20   common  UT:Eth0/0/2(U)
 TG:Eth0/0/1(U)
VID  Status  Property      MAC-LRN Statistics Description
1    enable  default       enable   disable   VLAN 0001
10   enable  default       enable   disable   VLAN 0010
20   enable  default       enable   disable   VLAN 0020

[S2]display vlan
The total number of vlans is : 3
--------------------------------------------------------------------------------
U: Up;          D: Down;          TG: Tagged;          UT: Untagged;
MP: Vlan-mapping;                 ST: Vlan-stacking;
#: ProtocolTransparent-vlan;      *: Management-vlan;
--------------------------------------------------------------------------------
VID  Type    Ports
--------------------------------------------------------------------------------
1    common  UT:Eth0/0/1(U)    Eth0/0/2(U)     Eth0/0/3(U)     Eth0/0/4(D)
                Eth0/0/5(D)    Eth0/0/6(D)     Eth0/0/7(D)     Eth0/0/8(D)
                Eth0/0/9(D)    Eth0/0/10(D)    Eth0/0/11(D)    Eth0/0/12(D)
                Eth0/0/13(D)   Eth0/0/14(D)    Eth0/0/15(D)    Eth0/0/16(D)
                Eth0/0/17(D)   Eth0/0/18(D)    Eth0/0/19(D)    Eth0/0/20(D)
                Eth0/0/21(D)   Eth0/0/22(D)    GE0/0/1(D)      GE0/0/2(D)
10   common  UT:Eth0/0/3(U)
             TG:Eth0/0/1(U)
20   common  UT:Eth0/0/2(U)
             TG:Eth0/0/1(U)
VID  Status  Property      MAC-LRN      Statistics   Description
1    enable  default       enable       disable      VLAN 0001
10   enable  default       enable       disable      VLAN 0010
20   enable  default       enable       disable      VLAN 0020
```

可以观察到，同样的需求，Hybrid 和 Access、Trunk 都能实现（测试省略），但 Hybrid 的灵活性及解决复杂需求的能力是 Access 和 Trunk 达不到的。

3. 实现网管员对所有网络的访问

在实现各部门内部终端可以互相访问，不同部门间的终端隔离访问后，要求网络管理员所在的 IT 部门（使用终端 PC-5）能够实现对所有部门的访问。即要求实现 VLAN 30 访问 VLAN 10 和 VLAN 20，VLAN 10 和 VLAN 20 之间仍然不允许互相访问。如果 S1

的E 0/0/2接口仍是Access类型且属于VLAN 10，则不能被其他VLAN访问。若要VLAN 30的终端能访问VLAN 10的终端，则需要修改接口的配置，使其既能被VLAN 10访问，又能被VLAN 30访问，这就要求此接口同时要属于多个VLAN，且端口所连设备是PC，不能识别带VLAN Tag的帧，故此时只能使用Hybrid类型接口。Hybrid端口既能被加入多个VLAN中，又能够在将其余VLAN的帧转发到此接口时，剥离掉相应的VLAN Tag。

配置S1交换机，E 0/0/4接口是网络管理员的PC终端，属于VLAN 30，该接口收到的PC发送的Untagged帧要能够发送至VLAN 30中，配置**port hybrid pvid vlan 30**命令设置Untagged帧加入至VLAN 30。

```
[S1]vlan 30
[S1-vlan30]interface Ethernet 0/0/4
[S1-Ethernet0/0/4]port hybrid pvid vlan 30
```

因为在华为交换上，默认所有接口都为Hybrid类型接口，所以在该接口下不需要修改配置。

S1交换机收到VLAN 10、VLAN 20和VLAN 30的帧也要能够从该接口发送至PC，配置**port hybrid untagged vlan 10 20 30**命令使得上述3个VLAN的帧会以Untagged的方式从该接口发送出去。

```
[S1-Ethernet0/0/4]port hybrid untagged vlan 10 20 30
```

同理，端口E 0/0/2接PC-1，接口收到PC的Untagged帧需要发送至VLAN 20，使用**port hybrid pvid vlan 20**命令。E 0/0/2接口同时也要能够被VLAN 30和VLAN 20的主机访问，即VLAN 20和30的帧能够从该接口发送出去，并以Untagged的方式发送至PC-1。

```
[S1]interface ethernet0/0/2
[S1-Ethernet0/0/2]port hybrid untagged vlan 20 30
```

接口E 0/0/3收到Untagged的帧需发送至VLAN 10，同时VLAN 10和30的帧要能从该接口发送出去。

```
[S1]interface ethernet0/0/3
[S1-Ethernet0/0/3]port hybrid untagged vlan 10 30
```

VLAN10、VLAN20和VLAN30的帧要能够发送至邻居交换机S2，且要保留原有的VLAN Tag，以便于邻居交换机S2根据VLAN Tag继续转发到相应的VLAN。同样，邻居交换机S2发送过来的帧也会带有相应的VLAN Tag，所以S1与S2间互连的接口E 0/0/1配置如下。

```
[S1]interface ethernet0/0/1
[S1-Ethernet0/0/1]port hybrid tagged vlan 10 20 30

[S2]interface ethernet0/0/1
[S2-Ethernet0/0/1]port hybrid tagged vlan 10 20 30
```

同理在S2交换机上，E 0/0/1接口收到的带有相应VLAN Tag标记的帧，如果是VLAN 10的帧要能发送至接口E 0/0/3，如果是VLAN 20的帧要能发送至E 0/0/2。而如果是VLAN 30的帧要能发送至接口E 0/0/2和E 0/0/3。VLAN10、20和30的帧都是以Untagged的方式发送至接口E 0/0/2或E 0/0/3。反之，如果PC-3发出的Untagged的帧发送至接口E 0/0/2时会进入到Hybrid接口PVID所指明的VLAN 20中，PC-4发出的Untagged的帧发送至接口E 0/0/3时会进入到Hybrid接口PVID所指明的VLAN 10中，具体配置过程如下。

```
[S2]interface ethernet0/0/2
[S2-Ethernet0/0/2]port hybrid untagged vlan 20 30
[S2-Ethernet0/0/2]interface ethernet 0/0/3
[S2-Ethernet0/0/3]port hybrid untagged vlan 10 30
```

S1 和 S2 上全部配置完成后，使用 **ping** 命令在 IT 部门的网络管理员的 PC-5 上测试与不同部门内的各台主机间的连通性，以 PC-1 为例。

```
PC>ping 192.168.1.1
Ping 192.168.1.1: 32 data bytes, Press Ctrl_C to break
From 192.168.1.1: bytes=32 seq=1 ttl=128 time=15 ms
From 192.168.1.1: bytes=32 seq=2 ttl=128 time<1 ms
From 192.168.1.1: bytes=32 seq=3 ttl=128 time<1 ms
From 192.168.1.1: bytes=32 seq=4 ttl=128 time=15 ms
From 192.168.1.1: bytes=32 seq=5 ttl=128 time=15 ms
--- 192.168.1.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 0/9/15 ms
```

可以观察到，PC-5 所属网络管理员所在的 VLAN 30，能够正常访问到其他部门的所有终端。

同理，选择市场部门所在 VLAN 20 内的主机 PC-1，测试其与其他主机间的连通性。

测试 PC-1 与本部门内的主机 PC-3 间的连通性。

```
PC>ping 192.168.1.3
Ping 192.168.1.3: 32 data bytes, Press Ctrl_C to break
From 192.168.1.3: bytes=32 seq=1 ttl=128 time=31 ms
From 192.168.1.3: bytes=32 seq=2 ttl=128 time=62 ms
From 192.168.1.3: bytes=32 seq=3 ttl=128 time=46 ms
From 192.168.1.3: bytes=32 seq=4 ttl=128 time=31 ms
From 192.168.1.3: bytes=32 seq=5 ttl=128 time=47 ms
--- 192.168.1.3 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 31/43/62 ms
```

可以正常通信。

测试 PC-1 与外部门的主机 PC-2 和 PC-4 间的连通性。

```
PC>ping 192.168.1.2
Ping 192.168.1.2: 32 data bytes, Press Ctrl_C to break
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
……

PC>ping 192.168.1.4
Ping 192.168.1.4: 32 data bytes, Press Ctrl_C to break
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
……
```

不能正常通信，实现了设计要求。

测试 PC-1 与 IT 部门网络管理员主机 PC-5 间的连通性。

```
PC>ping 192.168.1.100
Ping 192.168.1.100: 32 data bytes, Press Ctrl_C to break
From 192.168.1.100: bytes=32 seq=1 ttl=128 time<1 ms
From 192.168.1.100: bytes=32 seq=2 ttl=128 time=16 ms
From 192.168.1.100: bytes=32 seq=3 ttl=128 time<1 ms
From 192.168.1.100: bytes=32 seq=4 ttl=128 time<1 ms
From 192.168.1.100: bytes=32 seq=5 ttl=128 time<1 ms
--- 192.168.1.100 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 0/3/16 ms
```

可以正常通信。

在交换机上可以定义多个 VLAN，每个 VLAN 都可以看做是一个广播域，通常情况下每个 VLAN 都会分配一个独立的 IP 网络，根据需要把相应主机所在的接口划入到指定的 VLAN 中，并配置相应的网络 IP 地址，VLAN 间通过路由来实现互相访问。这是较为常用的方法。但是相比于基于端口的 Hybrid 配置，三层路由方式则不够灵活，原因在于 VLAN 之间的访问控制要借助于路由设备来实现。而控制 VLAN 访问使用 Hybrid 接口则极大地简化了配置的复杂性，它仅需在端口上自主定义基于 VLAN Tag 的过滤规则，来决定指定的 VLAN 的二层帧是否允许发送；它是通过二层来实现 VLAN 间的访问控制，既不需要每个 VLAN 定义单独的 IP 网段，更不需要在 VLAN 间引入路由设备，配置更为灵活方便。

思考

在本实验中，如果将 PC-5 所连交换机的接口 E 0/0/4 下的 **port hybrid pvid 30** 命令删除，PC-4 所连的端口 E 0/0/3 下 **port hybrid pvid 10** 命令删除，其他端口配置则保持不变。此时在 PC-5 与 PC-4 间的连通性是否正常？报文经过 S1 和 S2 间端口时使用的 VLAN Tag 是哪个？为什么？

3.4 利用单臂路由实现 VLAN 间路由

原理概述

以太网中，通常会使用 VLAN 技术隔离二层广播域来减少广播的影响，并增强网络的安全性和可管理性。其缺点是同时也严格地隔离了不同 VLAN 之间的任何二层流量，使分属于不同 VLAN 的用户不能直接互相通信。在现实中，经常会出现某些用户需要跨越 VLAN 实现通信的情况，单臂路由技术就是解决 VLAN 间通信的一种方法。

单臂路由的原理是通过一台路由器，使 VLAN 间互通数据通过路由器进行三层

转发。如果在路由器上为每个 VLAN 分配一个单独的路由器物理接口，随着 VLAN 数量的增加，必然需要更多的接口，而路由器能提供的接口数量比较有限，所以在路由器的一个物理接口上通过配置子接口（即逻辑接口）的方式来实现以一当多的功能，将是一种非常好的方式。路由器同一物理接口的不同子接口作为不同 VLAN 的默认网关，当不同 VLAN 间的用户主机需要通信时，只需将数据包发送给网关，网关处理后再发送至目的主机所在 VLAN，从而实现 VLAN 间通信。由于从拓扑结构图上看，在交换机与路由器之间，数据仅通过一条物理链路传输，故被形象地称之为“单臂路由”。

实验目的

- 理解单臂路由的应用场景
- 掌握路由器子接口的配置方法
- 掌握子接口封装 VLAN 的配置方法
- 理解单臂路由的工作原理

实验内容

本实验模拟公司网络场景。路由器 R1 是公司的出口网关，员工 PC 通过接入层交换机（如 S2 和 S3）接入公司网络，接入层交换机又通过汇聚交换机 S1 与路由器 R1 相连。公司内部网络通过划分不同的 VLAN 隔离了不同部门之间的二层通信，保证各部门间的信息安全，但是由于业务需要，经理、市场部和人事部之间需要能实现跨 VLAN 通信，网络管理员决定借助路由器的三层功能，通过配置单臂路由来实现。

实验拓扑

利用单臂路由实现 VLAN 间路由拓扑如图 3-5 所示。

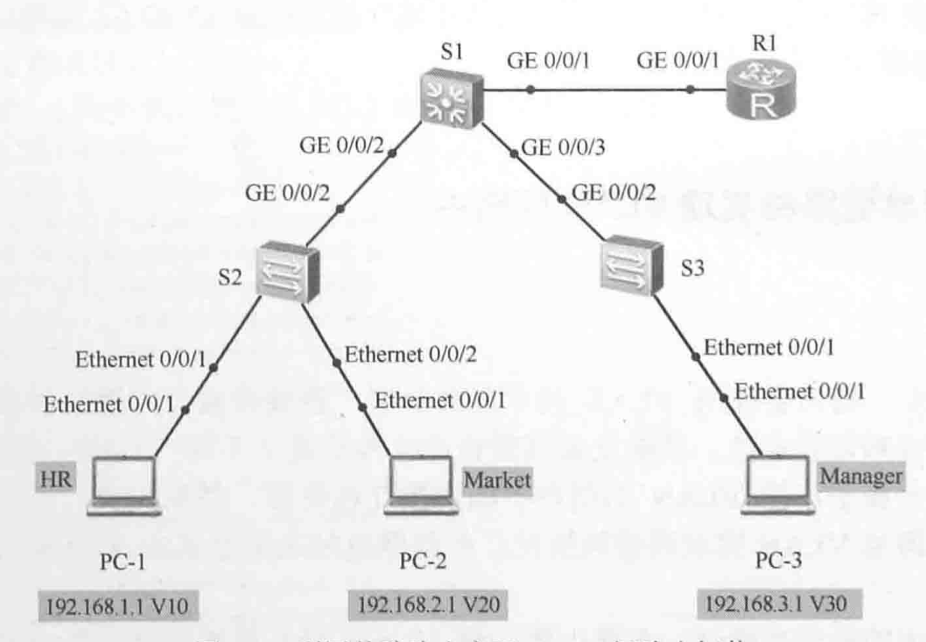


图 3-5 利用单臂路由实现 VLAN 间路由拓扑

实验编址

实验编址见表 3-4。

表 3-4 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR2220）	GE 0/0/1.1	192.168.1.254	255.255.255.0	N/A
	GE 0/0/1.2	192.168.2.254	255.255.255.0	N/A
	GE 0/0/1.3	192.168.3.254	255.255.255.0	N/A
PC-1	Ethernet 0/0/1	192.168.1.1	255.255.255.0	192.168.1.254
PC-2	Ethernet 0/0/1	192.168.2.1	255.255.255.0	192.168.2.254
PC-3	Ethernet 0/0/1	192.168.3.1	255.255.255.0	192.168.3.254

实验步骤

1. 创建 VLAN 并配置 Access、Trunk 接口

公司为保障各部门的信息安全，需保证隔离不同部门间的二层通信，规划各部门的终端属于不同的 VLAN，并为 PC 配置相应 IP 地址。

在 S2 上创建 VLAN 10 和 VLAN 20，把连接 PC-1 的 E 0/0/1 和连接 PC-2 的 E 0/0/2 接口配置为 Access 类型接口，并分别划分到相应的 VLAN 中。

```
[S2]vlan 10
[S2-vlan10]description HR
[S2-vlan10]vlan 20
[S2-vlan20]description Market
[S2-vlan20]interface Ethernet 0/0/1
[S2-Ethernet0/0/1]port link-type access
[S2-Ethernet0/0/1]port default vlan 10
[S2-Ethernet0/0/1]interface Ethernet 0/0/2
[S2-Ethernet0/0/2]port link-type access
[S2-Ethernet0/0/2]port default vlan 20
```

在 S3 上创建 VLAN 30，把连接 PC-3 的 E 0/0/1 接口配置为 Access 类型接口，并划分到 VLAN 30。

```
[S3]vlan 30
[S3-vlan30]description Manager
[S3-vlan30]interface Ethernet 0/0/1
[S3-Ethernet0/0/1]port link-type access
[S3-Ethernet0/0/1]port default vlan 30
```

交换机之间或交换机和路由器之间相连的接口需要传递多个 VLAN 信息，需要配置成 Trunk 接口。

将 S2 和 S3 的 GE 0/0/2 接口配置成 Trunk 类型接口，并允许所有 VLAN 通过。

```
[S2]interface GigabitEthernet 0/0/2
[S2-GigabitEthernet0/0/2]port link-type trunk
[S2-GigabitEthernet0/0/2]port trunk allow-pass vlan all
```

```
[S3]interface GigabitEthernet 0/0/2
[S3-GigabitEthernet0/0/2]port link-type trunk
[S3-GigabitEthernet0/0/2]port trunk allow-pass vlan all
```

在 S1 上创建 VLAN 10、VLAN 20 和 VLAN 30，并配置交换机和路由器相连的接口为 Trunk，允许所有 VLAN 通过。

```
[S1]vlan 10
[S1-vlan10]vlan 20
[S1-vlan20]vlan 30
[S1-vlan30]interface GigabitEthernet 0/0/2
[S1-GigabitEthernet0/0/2]port link-type trunk
[S1-GigabitEthernet0/0/2] port trunk allow-pass vlan all
[S1-GigabitEthernet0/0/2]interface GigabitEthernet 0/0/3
[S1-GigabitEthernet0/0/3]port link-type trunk
[S1-GigabitEthernet0/0/3] port trunk allow-pass vlan all
[S1-GigabitEthernet0/0/3]interface GigabitEthernet 0/0/1
[S1-GigabitEthernet0/0/1]port link-type trunk
[S1-GigabitEthernet0/0/1]port trunk allow-pass vlan all
```

2. 配置路由器子接口和 IP 地址

由于路由器 R1 只有一个实际的物理接口与交换机 S1 相连，可以在路由器上配置不同的逻辑子接口来作为不同 VLAN 的网关，从而达到节省路由器接口的目的。

在 R1 上创建子接口 GE 0/0/1.1，配置 IP 地址 192.168.1.254/24，作为人事部网关地址。

```
[R1]interface GigabitEthernet 0/0/1.1
[R1-GigabitEthernet0/0/1.1]ip address 192.168.1.254 24
```

在 R1 上创建子接口 GE 0/0/1.2，配置 IP 地址 192.168.2.254/24，作为市场部网关地址。

```
[R1]interface GigabitEthernet 0/0/1.2
[R1-GigabitEthernet0/0/1.2]ip address 192.168.2.254 24
```

在 R1 上创建子接口 GE 0/0/1.3，配置 IP 地址 192.168.3.254/24，作为经理的网关地址。

```
[R1]interface GigabitEthernet 0/0/1.3
[R1-GigabitEthernet0/0/1.3]ip address 192.168.3.254 24
```

在 PC-1、PC-2 和 PC-3 上配置 IP 和相应的网关地址后，在 PC-1 上测试与 PC-2 和 PC-3 间的连通性。

```
PC>ping 192.168.2.1
Ping 192.168.2.1: 32 data bytes, Press Ctrl_C to break
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
……

PC>ping 192.168.3.1
Ping 192.168.3.1: 32 data bytes, Press Ctrl_C to break
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
```

```
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
……
```

可以观察到，通信仍然无法建立。

3. 配置路由器子接口封装 VLAN

虽然目前已经创建了不同的子接口，并配置了相关 IP 地址，但是仍然无法通信。这是由于处于不同 VLAN 下，不同网段的 PC 间要实现互相通信，数据包必须通过路由器进行中转。由 S1 发送到 R1 的数据都加上了 VLAN 标签，而路由器作为三层设备，默认无法处理带了 VLAN 标签的数据包。因此需要在路由器上的子接口下配置对应 VLAN 的封装，使路由器能够识别和处理 VLAN 标签，包括剥离和封装 VLAN 标签。

在 R1 的子接口 GE 0/0/1.1 上封装 VLAN 10，在子接口 GE 0/0/1.2 上封装 VLAN 20，在子接口 GE 0/0/1.3 上封装 VLAN 30，并开启子接口的 ARP 广播功能。

使用 **dot1q termination vid** 命令配置子接口对一层 tag 报文的终结功能。即配置该命令后，路由器子接口在接收带有 VLAN tag 的报文时，将剥掉 tag 进行三层转发，在发送报文时，会将与该子接口对应 VLAN 的 VLAN tag 添加到报文中。

```
[R1-GigabitEthernet0/0/1.1]dot1q termination vid 10
```

使用 **arp broadcast enable** 命令开启子接口的 ARP 广播功能。如果不配置该命令，将会导致该子接口无法主动发送 ARP 广播报文，以及向外转发 IP 报文。

```
[R1-GigabitEthernet0/0/1.1]arp broadcast enable
```

同理配置 R1 的子接口 GE 0/0/1.2 和 GE 0/0/1.3。

```
[R1]interface GigabitEthernet0/0/1.2
[R1-GigabitEthernet0/0/1.2]dot1q termination vid 20
[R1-GigabitEthernet0/0/1.2]arp broadcast enable
[R1-GigabitEthernet0/0/1.2]interface GigabitEthernet0/0/1.3
[R1-GigabitEthernet0/0/1.3]dot1q termination vid 30
[R1-GigabitEthernet0/0/1.3]arp broadcast enable
```

配置完成后，在路由器 R1 上查看接口状态。

```
[R1]display ip interface brief
*down: administratively down
……
Interface                   IP Address/Mask      Physical   Protocol
GigabitEthernet0/0/0          unassigned         down       down
GigabitEthernet0/0/1          unassigned         up         down
GigabitEthernet0/0/1.1        192.168.1.254/24   up         up
GigabitEthernet0/0/1.2        192.168.2.254/24   up         up
GigabitEthernet0/0/1.3        192.168.3.254/24   up         up
GigabitEthernet0/0/2          unassigned         down       down
NULL0                         unassigned         up         up (s)
```

可以观察到，3 个子接口的物理状态和协议状态都正常。

查看路由器 R1 的路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
```

```
Routing Tables: Public
         Destinations : 13        Routes : 13
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
       127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
       127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
127.255.255.255/32   Direct  0    0           D   127.0.0.1       InLoopBack0
    192.168.1.0/24   Direct  0    0           D   192.168.1.254   GigabitEthernet0/0/1.1
  192.168.1.254/32   Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/1.1
  192.168.1.255/32   Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/1.1
    192.168.2.0/24   Direct  0    0           D   192.168.2.254   GigabitEthernet0/0/1.2
  192.168.2.254/32   Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/1.2
  192.168.2.255/32   Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/1.2
    192.168.3.0/24   Direct  0    0           D   192.168.3.254   GigabitEthernet0/0/1.3
  192.168.3.254/32   Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/1.3
  192.168.3.255/32   Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/1.3
255.255.255.255/32   Direct  0    0           D   127.0.0.1       InLoopBack0
```

可以观察到，路由表中已经有了 192.168.1.0/24、192.168.2.0/24、192.168.3.0/24 的路由条目，并且都是路由器 R1 的直连路由，类似于路由器上的直连物理接口。

在 PC-1 上分别测试与网关地址 192.168.1.254 和 PC-2 间的连通性。

```
PC>ping 192.168.1.254
Ping 192.168.1.254: 32 data bytes, Press Ctrl_C to break
From 192.168.1.254: bytes=32 seq=1 ttl=255 time=62 ms
From 192.168.1.254: bytes=32 seq=2 ttl=255 time=62 ms
From 192.168.1.254: bytes=32 seq=3 ttl=255 time=46 ms
From 192.168.1.254: bytes=32 seq=4 ttl=255 time=32 ms
From 192.168.1.254: bytes=32 seq=5 ttl=255 time=47 ms
--- 192.168.1.254 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 32/49/62 ms

PC>ping 192.168.2.1
Ping 192.168.2.1: 32 data bytes, Press Ctrl_C to break
From 192.168.2.1: bytes=32 seq=1 ttl=127 time=94 ms
From 192.168.2.1: bytes=32 seq=2 ttl=127 time=78 ms
From 192.168.2.1: bytes=32 seq=3 ttl=127 time=109 ms
From 192.168.2.1: bytes=32 seq=4 ttl=127 time=93 ms
From 192.168.2.1: bytes=32 seq=5 ttl=127 time=110 ms
--- 192.168.2.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 78/96/110 ms
```

可以观察到，通信正常。在 PC-1 上 Tracert　PC-2。

```
PC>tracert 192.168.2.1
traceroute to 192.168.2.1, 8 hops max
(ICMP), press Ctrl+C to stop
1   192.168.1.254    62 ms    47 ms   31 ms
```

```
2  192.168.2.1      125 ms  94 ms  94 ms
```

可以观察到 PC-1 先把 ping 包发送给自身的网关 192.168.1.254，然后再由网关发送到 PC-2。

现以 PC-1 ping PC-2 为例，分析单臂路由的整个运作过程。

两台 PC 由于处于不同的网络中，这时 PC-1 会将数据包发往自己的网关，即路由器 R1 的子接口 GE 0/0/1.1 的地址 192.168.1.254。

数据包到达路由器 R1 后，由于路由器的子接口 GE 0/0/1.1 已经配置了 VLAN 封装，当接收到 PC-1 发送的 VLAN 10 的数据帧时，发现数据帧的 VLAN ID 跟自身 GE 0/0/1.1 接口配置的 VLAN ID 一样，便会剥离掉数据帧的 VLAN 标签后通过三层路由转发。

通过查找路由表后，发现数据包中的目的地址 192.168.2.1 所属的 192.168.2.0/24 网段的路由条目，已经是路由器 R1 上的直连路由，且出接口为 GE 0/0/1.2，便将该数据包发送至 GE 0/0/1.2 接口。

当 GE 0/0/1.2 接口接收到一个没有带 VLAN 标签的数据帧时，便会加上自身接口所配置的 VLAN ID 20 后再进行转发，然后通过交换机将数据帧顺利转发给 PC-2。

思考

VLAN 间的通信可以利用单臂路由的方式实现，那么利用单臂路由实现数据转发会存在哪些潜在问题？该如何解决？

3.5 利用三层交换机实现 VLAN 间路由

原理概述

VLAN 将一个物理的 LAN 在逻辑上划分成多个广播域。VLAN 内的主机间可以直接通信，而 VLAN 间不能直接互通。

在现实网络中，经常会遇到需要跨 VLAN 相互访问的情况，工程师通常会选择一些方法来实现不同 VLAN 间主机的相互访问，例如单臂路由。但是单臂路由技术中由于存在一些局限性，比如带宽、转发效率等，使得这项技术应用较少。

三层交换机在原有二层交换机的基础之上增加了路由功能，同时由于数据没有像单臂路由那样经过物理线路进行路由，很好地解决了带宽瓶颈的问题，为网络设计提供了一个灵活的解决方案。

VLANIF 接口是基于网络层的接口，可以配置 IP 地址。借助 VLANIF 接口，三层交换机就能实现路由转发功能。

实验目的

- 掌握配置 VLANIF 接口的方法
- 理解数据包跨 VLAN 路由的原理

- 掌握测试多层交换网络连通性的方法

实验内容

本实验模拟企业网络场景。公司有两个部门——销售部和客服部，分别规划使用 VLAN 10 和 VLAN 20。其中销售部下有两台终端 PC-1 和 PC-2，客服部下有一台终端 PC-3。所有终端都通过核心三层交换机 S1 相连。现需要让该公司所有三台主机都能实现互相访问，网络管理员将通过配置三层交换机来实现。

实验拓扑

利用三层交换机实现 VLAN 间路由的拓扑如图 3-6 所示。

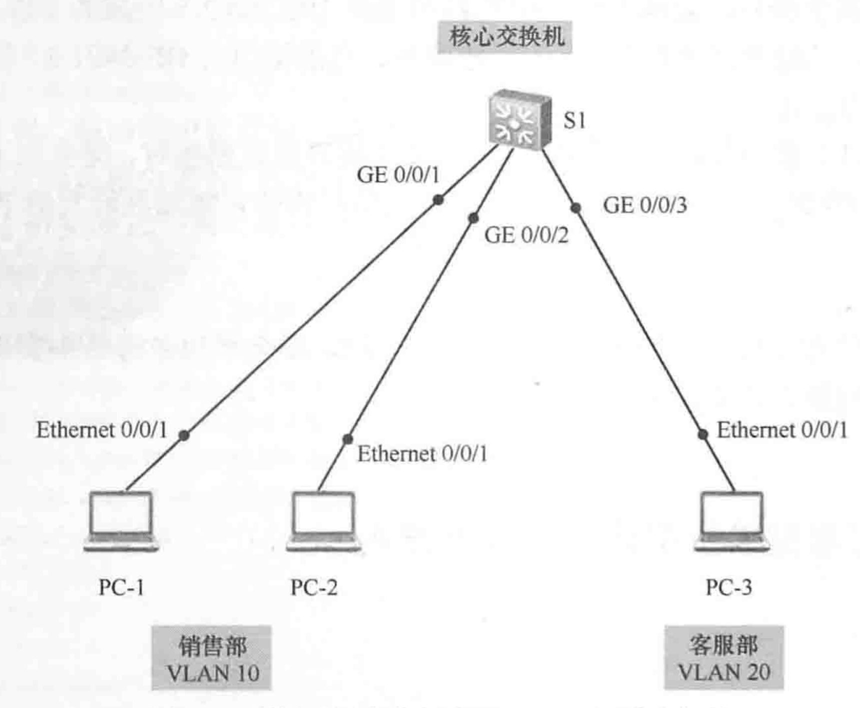


图 3-6 利用三层交换机实现 VLAN 间路由拓扑

实验编址

实验编址见表 3-5。

表 3-5 实验编址

设备	接口	IP 地址	子网掩码	网关
PC-1	Ethernet 0/0/1	192.168.1.1	255.255.255.0	192.168.1.254
PC-2	Ethernet 0/0/1	192.168.1.2	255.255.255.0	192.168.1.254
PC-3	Ethernet 0/0/1	192.168.2.1	255.255.255.0	192.168.2.254
S1（S5700）	VLANIF 10	192.168.1.254	255.255.255.0	N/A
	VLANIF 20	192.168.2.254	255.255.255.0	N/A

实验步骤

1. 基本配置

根据实验编址表在 PC 上进行相应的基本 IP 地址配置，三层交换机 S1 上暂先不做配置。

配置完成后，测试销售部两台终端 PC-1 与 PC-2 间的连通性。

```
PC>ping 192.168.1.1
Ping 192.168.1.1: 32 data bytes, Press Ctrl_C to break
From 192.168.1.1: bytes=32 seq=1 ttl=128 time=31 ms
From 192.168.1.1: bytes=32 seq=2 ttl=128 time=15 ms
From 192.168.1.1: bytes=32 seq=3 ttl=128 time<1 ms
From 192.168.1.1: bytes=32 seq=4 ttl=128 time<1 ms
From 192.168.1.1: bytes=32 seq=5 ttl=128 time<1 ms
--- 192.168.1.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 0/9/31 ms
```

可以观察到，通信正常。

测试销售部 PC-1 与客服部 PC-3 间的连通性。

```
PC>ping 192.168.2.1
Ping 192.168.2.1: 32 data bytes, Press Ctrl_C to break
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
......
```

PC-1 与 PC-3 间无法正常通信，下面简要分析主机 PC-1 发出数据包，直至反馈目的无法到达的整个过程：

主机发出数据包前，将会查看数据包中的目的 IP 地址，如果目的 IP 地址和本机 IP 地址在同一个网段上，主机会直接发出一个 ARP 请求数据包来请求对方主机的 MAC 地址，封装数据包，继而发送该数据包。但如果目的 IP 地址与本机 IP 地址不在同一个网段，那么主机也会发出一个 ARP 数据包请求网关的 MAC 地址，收到网关 ARP 回复后，继而封装数据包后发送。

所以，销售部主机 PC-1 在访问 192.168.2.1 这个 IP 地址时发现这个目的 IP 地址与本机 IP 地址不在同一个 IP 地址段上，PC-1 便会发出 ARP 数据包请求网关 192.168.1.254 的 MAC 地址。但由于交换机没有做任何 IP 配置，因此没有设备应答该 ARP 请求，导致销售部主机 PC-1 无法正常封装数据包，因此无法与客服部 PC-3 正常通信。

2. 配置三层交换机实现 VLAN 间通信

通过在交换机上设置不同的 VLAN 使得主机实现相互隔离。在三层交换机 S1 上创建 VLAN 10 和 VLAN 20，把销售部的主机全部划入 VLAN 10 中，客服部的主机划入

VLAN 20 中。

```
[S1]vlan 10
[S1-vlan10]vlan 20
[S1-vlan20]interface GigabitEthernet0/0/1
[S1-GigabitEthernet0/0/1]port link-type access
[S1-GigabitEthernet0/0/1]port default VLAN 10
[S1-GigabitEthernet0/0/1]interface GigabitEthernet0/0/2
[S1-GigabitEthernet0/0/2]port link-type access
[S1-GigabitEthernet0/0/2]port default VLAN 10
[S1-GigabitEthernet0/0/2]interface GigabitEthernet0/0/3
[S1-GigabitEthernet0/0/3]port link-type access
[S1-GigabitEthernet0/0/3]port default VLAN 20
```

现在需要通过 VLAN 间路由来实现通信，在三层交换机上配置 VLANIF 接口。

在 S1 上使用 **interface VLANif** 命令创建 VLANIF 接口，指定 VLANIF 接口所对应的 VLAN ID 为 10，并进入 VLANIF 接口视图，在接口视图下配置 IP 地址 192.168.1.254/24。再创建对应 VLAN 20 的 VLANIF 接口，地址配置为 192.168.2.254/24。

```
[S1]interface VLANif 10
[S1-VLANif10]ip address 192.168.1.254 24
[S1-VLANif10]interface VLANif 20
[S1-VLANif20]ip address 192.168.2.254 24
```

配置完成后，查看接口状态。

```
[S1]display ip interface brief
*down: administratively down
……
Interface                IP Address/Mask        Physical    Protocol
MEth0/0/1                  unassigned           down        down
NULL0                      unassigned           up          up(s)
VLANif1                    unassigned           down        down
VLANif10                   192.168.1.254/24     up          up
VLANif20                   192.168.2.254/24     up          up
```

可以观察到，两个 VLANIF 接口已经生效。再次测试 PC-1 与 PC-3 间的连通性。

```
PC>ping 192.168.2.1
Ping 192.168.2.1: 32 data bytes, Press Ctrl_C to break
From 192.168.2.1: bytes=32 seq=1 ttl=127 time=-61765 ms
From 192.168.2.1: bytes=32 seq=2 ttl=127 time=-61781 ms
From 192.168.2.1: bytes=32 seq=3 ttl=127 time=-61766 ms
From 192.168.2.1: bytes=32 seq=4 ttl=127 time=-61766 ms
From 192.168.2.1: bytes=32 seq=5 ttl=127 time=-61781 ms
--- 192.168.2.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = -61781/858931687/-61765 ms
```

可见通信正常，实现了销售部终端与客服部终端间的通信。PC-2 上的测试省略。

在 PC-1 上查看 ARP 信息。

```
PC>arp -a
```

```
Internet Address        Physical Address        Type
192.168.1.254           4C-1F-CC-63-AB-09       dynamic
```

可以观察到，目前 PC 上 ARP 解析到的地址只有交换机的 VLANIF 10 的地址，而没有对端的地址，PC-1 先将数据包发送至网关，即对应的 VLANIF 10 接口，再由网关转发到对端。

思考

试问三层交换机与路由器实现三层功能的方式是否相同？为什么？

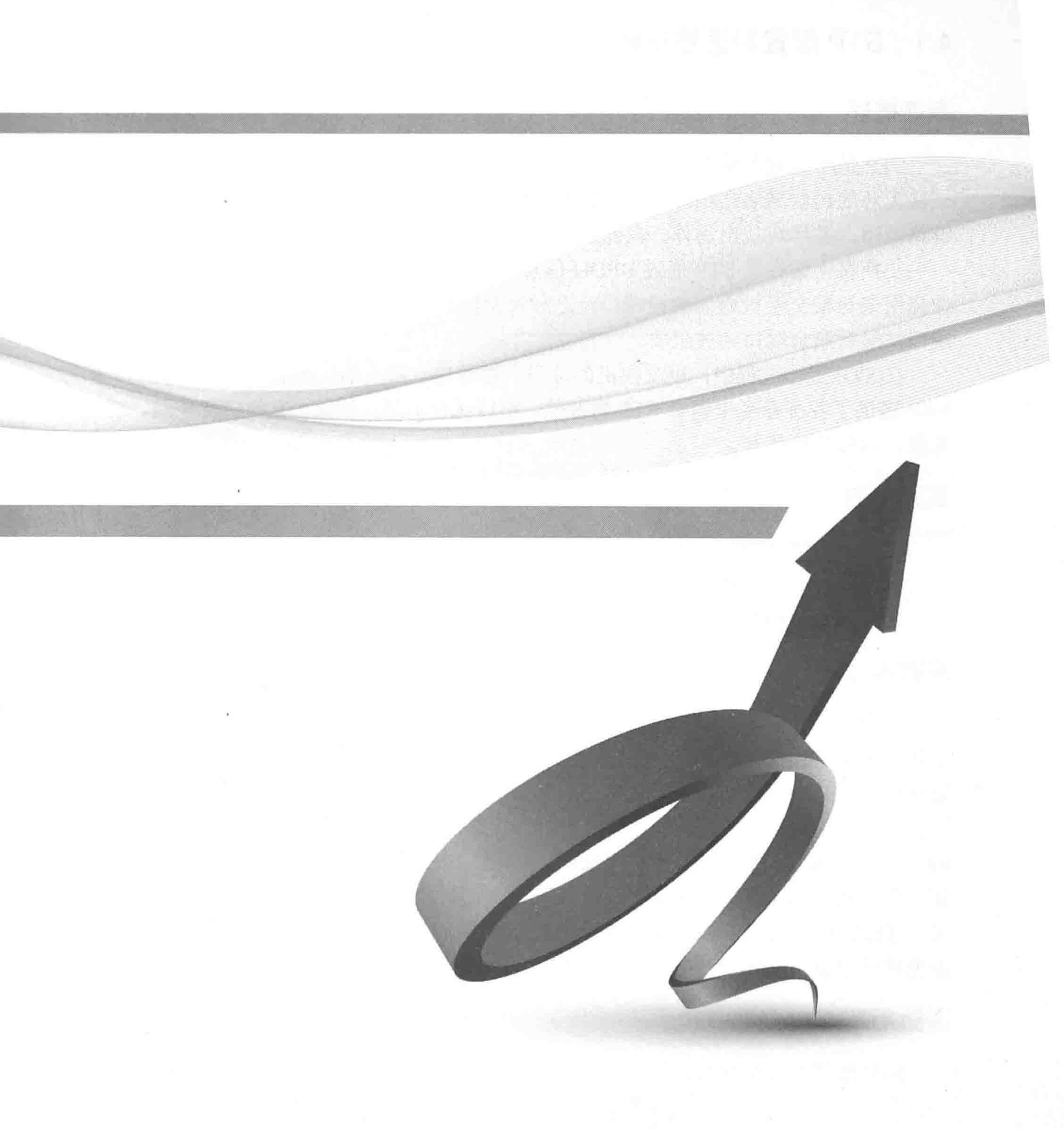
第4章 生成树

4.1 STP配置和选路规则

4.2 配置STP定时器

4.3 RSTP基础配置

4.4 MSTP基础配置



4.1 STP 配置和选路规则

原理概述

STP 是用来避免数据链路层出现逻辑环路的协议，使用 BPDU 传递网络信息计算出一根无环的树状网络结构，并阻塞特定端口。在网络出现故障的时候，STP 能快速发现链路故障，并尽快找出另外一条路径进行数据传输。

交换机上运行的 STP 通过 BPDU 信息的交互，选举根交换机，然后每台非根交换机选择用来与根交换机通信的根端口，之后每个网段选择用来转发数据至根交换机的指定端口，最后剩余端口则被阻塞。

在 STP 工作过程中，根交换机的选举，根端口、指定端口的选举都非常重要。华为 VRP 提供了各种命令来调整 STP 的参数，用以优化网络。例如，交换机优先级、端口优先级、端口代价值等。

实验目的

- 理解 STP 的选举过程
- 掌握修改交换机优先级的方法
- 掌握修改端口开销值的方法

实验内容

公司购置了 4 台交换机，组建网络。考虑到网络的可靠性，将 4 台交换机如图 4-1 所示拓扑搭建。由于默认情况下，交换机之间运行 STP 后，根交换机、根端口、指定端口的选择将基于交换机的 MAC 地址的大小，因此带来了不确定性，极可能由此产生隐患。

公司网络规划，需要 S1 作为主根交换机，S2 作为 S1 的备份根交换机。同时对于 S4 交换机，E 0/0/1 接口应该作为根端口。对于 S2 和 S3 之间的链路，应该保证 S2 的 E 0/0/3 接口作为指定端口。同时在交换机 S3 上，存在两个接口 E 0/0/10、E 0/0/11 连接到测试 PC，测试 PC 经常上下线网络，需要将交换机 S3 与之相连的对应端口定义为边缘端口，避免测试电脑上下线对网络产生的影响。

实验拓扑

STP 配置及选举规则的拓扑如图 4-1 所示。

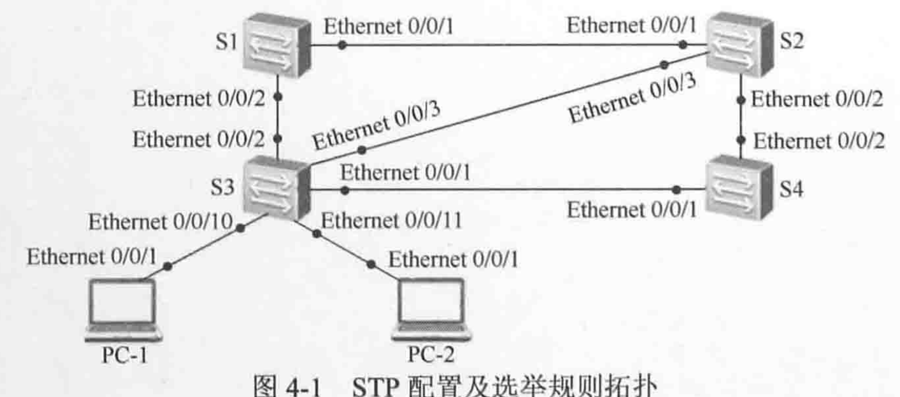


图 4-1 STP 配置及选举规则拓扑

MAC 地址

本实验的 MAC 地址见表 4-1。

表 4-1　　MAC 地址

设备	全局 MAC 地址
S1（S3700）	4c1f-cceb-beac
S2（S3700）	4c1f-ccbf-cbb5
S3（S3700）	4c1f-ccb0-58df
S4（S3700）	4c1f-ccac-3733

实验步骤

1. 基本配置

根据图 4-1，在交换机上启用 STP（华为交换机默认启用 MSTP），将交换机的 STP 模式更改为普通生成树 STP。

```
[S1]stp enable
[S1]stp mode stp

[S2]stp enable
[S2]stp mode stp

[S3]stp enable
[S3]stp mode stp

[S4]stp enable
[S4]stp mode stp
```

配置完成后，默认情况下需要等待 30s 生成树重新计算的时间（15s Forward Delay 加上 15s Learning 状态时间），再使用 **display stp** 命令查看 S1 的生成树状态。

```
[S1]display stp
-------[CIST Global Info][Mode STP]-------
CIST Bridge              :32768.4c1f-cceb-beac
……
Last TC occurred         :Ethernet0/0/1
----[Port1(Ethernet0/0/1)][FORWARDING]----
 Port Protocol           :Enabled
 Port Role               :Root Port
 Port Priority           :128
 Port Cost(Dot1T )       :Config=auto / Active=1
 Designated Bridge/Port  :32768.4c1f-ccbf-cbb5 / 128.1
……
BPDU Received            :50
          TCN: 0, Config: 50, RST: 0, MST: 0
----[Port2(Ethernet0/0/2)][DISCARDING]----
 Port Protocol           :Enabled
 Port Role               :Alternate Port
 Port Priority           :128
 Port Cost(Dot1T )       :Config=auto / Active=1
 Designated Bridge/Port  :32768.4c1f-cceb-658f / 128.2
……
```

可以观察到 S1 的 E 0/0/1 端口为转发状态、端口角色为根端口，E 0/0/2 端口为丢弃状态，端口角色 Alternate，即替代端口。

还可以使用 **display stp brief** 命令在 S2、S3、S4 上仅查看摘要信息。

```
[S2]display stp brief
 MSTID  Port                        Role   STP State        Protection
 0      Ethernet0/0/1               DESI   FORWARDING       NONE
 0      Ethernet0/0/2               ROOT   FORWARDING       NONE
 0      Ethernet0/0/3               DESI   FORWARDING       NONE
```

在交换机 S2 上所有的端口为转发状态，观察到 E 0/0/1 和 E 0/0/3 端口角色为指定端口，E 0/0/2 为根端口。

```
[S3]display stp brief
 MSTID  Port                        Role   STP State        Protection
    0   Ethernet0/0/1               ROOT   FORWARDING       NONE
    0   Ethernet0/0/2               DESI   FORWARDING       NONE
    0   Ethernet0/0/3               ALTE   DISCARDING       NONE
    ......
```

在交换机 S3 上 E 0/0/3 端口角色为 Alternate 端口，且状态为丢弃状态，该端口将不会转发数据流量。

```
[S4]display stp brief
 MSTID  Port                        Role   STP State        Protection
 0      Ethernet0/0/1               DESI   FORWARDING       NONE
 0      Ethernet0/0/2               DESI   FORWARDING       NONE
```

在交换机 S4 上所有的端口角色都为指定端口，且端口状态都为转发。

可以初步判断 4 台交换机中 S4 为根交换机，因为该交换机所有端口都为指定端口。通过 **display stp** 命令查看生成树详细信息。

```
[S4]display stp
-------[CIST Global Info][Mode STP]-------
CIST Bridge              :32768.4c1f-ccac-3733
Config Times             :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
Active Times             :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
CIST Root/ERPC           :32768.4c1f-ccac-3733 / 0
CIST RegRoot/IRPC        :32768.4c1f-ccac-3733 / 0
......
```

可以观察到“CIST Root”和“CIST Bridge”相同，即目前根交换机 ID 与自身的交换机 ID 相同，说明目前 S4 为根交换机。

生成树运算第一步就是通过比较每台交换机的 ID 选举根交换机。交换机 ID 由交换机优先级和 MAC 地址组成，首先比较交换机优先级，数值最低的为根交换机；如果优先级一样，则比较 MAC 地址，同样数值最低的选举为根交换机。

目前在该公司的二层拓扑中，4 台交换机的生成树都刚刚开始运行，交换机优先级都为默认值，即都相同，故根据每台交换机的 MAC 地址来选举，通过比较，最终确定 S4 为根交换机。

2. 配置网络中的根交换机

根交换机在网络中的位置是非常重要的，如果选择了一台性能较差的交换机，或者是部署在接入层的交换机作为根交换机，会影响到整个网络的通信质量及数据传输。所以确定根交换机的位置极为重要。根交换机选举依据是根交换机 ID，值越小越优先，交换机默认的优先级为 32768，当然该值是可以修改的。

现在将 S1 配置为主根交换机，S2 为备份根交换机，将 S1 的优先级改为 0，S2 的优先级改为 4096。

```
[S1]stp priority 0

[S2]stp priority 4096
```

配置完成后查看 S1 和 S2 的 STP 状态信息。

```
[S1]display stp
-------[CIST Global Info][Mode STP]-------
CIST Bridge            :0     .4c1f-cceb-beac
Config Times           :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
Active Times           :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
CIST Root/ERPC         :0     .4c1f-cceb-beac / 0
CIST RegRoot/IRPC      :0     .4c1f-cceb-beac / 0
……

[S2]display stp
-------[CIST Global Info][Mode STP]-------
CIST Bridge            :4096 .4c1f-ccbf-cbb5
Config Times           :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
Active Times           :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
CIST Root/ERPC         :0     .4c1f-cceb-beac / 1
CIST RegRoot/IRPC      :4096 .4c1f-ccbf-cbb5 / 0
……
```

通过观察发现 S1 的优先级变为了 0，为根交换机；而 S2 的优先级变为了 4096，为备份根交换机。

这里还可以使用另外一种方式配置主根交换机和备份根交换机。

首先删除在 S1 上所配置的优先级，使用 **stp root primary** 命令配置主根交换机。

```
[S1]undo stp priority

[S1]stp root primary
```

删除在 S2 上所配置的优先级，使用 **stp root secondary** 命令配置备份根交换机。

```
[S1]undo stp priority

[S2]stp root secondary
```

配置完成后查看 STP 的状态信息，与前一种方法得到的一致，此时 S1 自动更改优先级为 0，而 S2 更改为 4096。

3. 理解根端口的选举

生成树在选举出根交换机之后，将在每台非根交换机上选举根端口。选举时首先比较该交换机上每个端口到达根交换机的根路径开销，路径开销最小的端口将成为根端口。如果根路径开销值相同，则比较每个端口所在链路上的上行交换机 ID，如果该交换机 ID 也相同，则比较每个端口所在链路上的上行端口 ID。每台交换机上只能拥有一个根端口。

目前 S1 为主根交换机，而 S2 为备份根交换机，查看 S4 上生成树信息。

```
[S4]display stp brief
 MSTID  Port                    Role  STP State     Protection
   0    Ethernet0/0/1           ALTE  DISCARDING      NONE
   0    Ethernet0/0/2           ROOT  FORWARDING      NONE
```

可以观察到，现在 S4 的 E 0/0/2 为根端口，状态为转发状态。S4 在选举根端口时，首先比较根路径开销，由于拓扑中所有链路都是相同的百兆以太网链路，S4 经过 S3 到

S1 与经过 S2 到 S1 的开销值相同；接下来比较 S4 的两台上行链路的交换机 S2 和 S3 的交换机标识，S2 目前的交换机优先级为 4096，而 S3 为默认的 32768，所以与 S2 连接的 E 0/0/2 接口被选为根端口。

查看 S4 的 E 0/0/2 接口开销值。

```
<S4>display stp interface Ethernet 0/0/2
----[Port2(Ethernet0/0/2)][FORWARDING]----
 Port Protocol          :Enabled
 Port Role              :Root Port
 Port Priority          :128
 Port Cost(Dot1T )      :Config=auto / Active=1
 Designated Bridge/Port     :4096.4c1f-ccbf-cbb5/ 128.2
......
```

可以观察到，接口路径开销采用的是 Dot1T 的计算方法，Config 是指手工配置的路径开销，Active 是实际使用的路径开销，开销值为 1。

配置 S4 的 E 0/0/2 接口的代价值为 2000，即增加该接口默认的代价值。

```
[S4]interface ethernet0/0/2
[S4-Ethernet0/0/2]stp cost 2000
```

配置完成后再次查看 S4 的 E 0/0/2 接口开销值以及 STP 状态摘要信息。

```
<S4>display stp interface Ethernet 0/0/2
----[Port2(Ethernet0/0/2)][DISCARDING]----
 Port Protocol          :Enabled
 Port Role              :Alternate Port
 Port Priority          :128
 Port Cost(Dot1T )      :Config=2000 / Active=2000
 Designated Bridge/Port     :4096.4c1f-ccbf-cbb5 / 128.2
 ......

[S4]display stp brief
 MSTID  Port                    Role  STP State     Protection
   0    Ethernet0/0/1           ROOT  FORWARDING      NONE
   0    Ethernet0/0/2           ALTE  DISCARDING      NONE
```

发现此时 E 0/0/1 端口角色变成了根端口，而 E 0/0/2 变成了 Alternate 端口。这是由于将 E 0/0/2 接口的开销修改为 2000 之后，在选举根端口时，其到根路径开销大于 E 0/0/1 的根路径开销。

4. 理解指定端口的选举

生成树协议在每台非根交换机选举出根端口之后，将在每个网段上选举指定端口，选举的比较规则和选举根端口类似。

现在网络管理员需要确保 S2 连接 S3 的 E 0/0/3 接口被选择为指定端口，可以通过修改端口开销值来实现。

为了模拟该场景，将 S2 的优先级恢复为默认的 32768。

```
[S2]undo stp root
```

配置完成后，查看 S2 的 STP 状态信息。

```
[S2]display stp
-------[CIST Global Info][Mode STP]-------
CIST Bridge             :32768. 4c1f-ccbf-cbb5
Config Times            :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
......
```

查看 S2 与 S3 的 STP 状态摘要信息。

```
[S2]display stp brief
 MSTID  Port                    Role   STP State      Protection
   0    Ethernet0/0/1           ROOT   FORWARDING      NONE
   0    Ethernet0/0/2           DESI   FORWARDING      NONE
   0    Ethernet0/0/3           ALTE   DISCARDING      NONE

[S3]display stp brief
 MSTID  Port                    Role   STP State      Protection
   0    Ethernet0/0/1           DESI   FORWARDING      NONE
   0    Ethernet0/0/2           ROOT   FORWARDING      NONE
   0    Ethernet0/0/3           DESI   FORWARDING      NONE
   0    Ethernet0/0/10          DESI   FORWARDING      NONE
   0    Ethernet0/0/11          DESI   FORWARDING      NONE
```

通过观察发现在 S2 与 S3 间的链路上，选择了 S3 的 E 0/0/3 接口为指定端口，而 S2 的 E 0/0/3 接口为 Alternate 端口。这是由于在选举指定端口时，首先比较两个端口的根路径开销，目前都相同；接着比较上行交换机的 ID，此时 S2 和 S3 的交换机优先级相同，故比较 MAC 地址，最后通过比较 MAC 地址得出。

查看 S2 和 S3 的 E 0/0/3 接口信息。

```
<S2>display interface Ethernet 0/0/3
Ethernet0/0/3 current state : UP
......
Current system time: 2013-08-30 13:48:06-08:00
Hardware address is 4c1f-ccbf-cbb5
    Last 300 seconds input rate 0 bytes/sec, 0 packets/sec
......

<S3>display  interface  Ethernet  0/0/3
Ethernet0/0/3 current state : UP
......
Current system time: 2013-08-30 13:49:15-08:00
Hardware address is 4c1f-ccb0-58df
    Last 300 seconds input rate 0 bytes/sec, 0 packets/sec
......
```

可以观察到，S2 上 E 0/0/3 接口的 MAC 地址大于 S3 上 E 0/0/3 接口的 MAC 地址，所以该网段上 S3 的 E 0/0/3 接口成为指定接口。

修改 S3 的 E 0/0/2 接口的开销值，将该值增大（默认为 1），即增大该端口上的根路径开销，确保让 S2 的 E 0/0/3 接口成为指定端口。

```
[S3]interface Ethernet 0/0/2
[S3-Ethernet0/0/2]stp cost 2
```

配置完成后查看 S2 的 STP 状态摘要信息。

```
[S2]display stp brief
 MSTID  Port                    Role   STP State      Protection
   0    Ethernet0/0/1           ROOT   FORWARDING      NONE
   0    Ethernet0/0/2           DESI   FORWARDING      NONE
   0    Ethernet0/0/3           DESI   FORWARDING      NONE
```

根据 STP 计算规则选择指定端口时，最终选择 S2 的 E 0/0/3 接口作为指定端口。

为了验证现在能够确保 S2 的 E 0/0/3 接口成为指定端口，下面将 S3 的优先级调整为 4096，并查看。

```
[S3]stp priority 4096

[S3]display stp
-------[CIST Global Info][Mode STP]-------
CIST Bridge            :4096. 4c1f-ccb0-58df
Config Times           :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
……
```

再次查看 S2 和 S3 的 STP 状态。

```
[S2]display stp brief
 MSTID  Port                    Role   STP State      Protection
   0    Ethernet0/0/1           ROOT   FORWARDING       NONE
   0    Ethernet0/0/2           DESI   FORWARDING       NONE
   0    Ethernet0/0/3           DESI   FORWARDING       NONE

[S3]display stp brief
 MSTID  Port                    Role   STP State      Protection
   0    Ethernet0/0/1           DESI   FORWARDING       NONE
   0    Ethernet0/0/2           ROOT   FORWARDING       NONE
   0    Ethernet0/0/3           ALTE   DISCARDING       NONE
……
```

可以观察到，即使将 S3 的优先级修改得比 S2 的优先级值更低，但是 S2 的 E 0/0/3 接口仍然为指定端口，而 S3 的 E 0/0/3 接口还是 Alternate 端口，再次验证了在选举指定端口时首先比较根路径开销的规则。

思考

在什么场景下，选举根端口、指定端口时会比较到端口 ID？

4.2 配置 STP 定时器

原理概述

普通生成树 STP 不能实现快速收敛，但是在 STP 中诸如 Hello Time 定时器、Max Age 定时器、Forward Delay 定时器、未收到上游的 BPDU 就重新开始生成树计算的超时时间等参数会影响其收敛速度。通过配置合适的系统参数，可以使 STP 实现最快的拓扑收敛。下面首先介绍 STP 定时器。

■ Hello Time 定时器：Hello Time 为周期发送 BPDU 来维护生成树的稳定的时间，默认为 2s。如果交换机在配置的超时时间内没有收到上游交换机发送的 BPDU，则会重新进行生成树计算。在根交换机上配置的 Hello Time 将作为整个生成树内所有交换机的 Hello Time。

■ Max Age 定时器：BPDU 的最大生存时间，默认为 20s，交换机通过比较从上游交换机收到的 BPDU 中携带的 Message Age（配置 BPDU 的生存时间，如果配置 BPDU 是根桥发出的，则 Message Age 为 0，每经过一台交换机增加 1）和 Max Age，来判断此 BPDU 是否超时。如果收到的 BPDU 超时，交换机将该 BPDU 老化，同时

阻塞接收该 BPDU 的接口，并开始发出以自己为根桥的 BPDU。这种老化机制可以有效地控制生成树的半径。在根交换机上配置的 Max Age 将作为整个生成树内所有交换机的 Max Age。

■ Forward Delay 定时器：此延迟时间为 Forward Delay 定时器的时间，默认为 15s。链路故障会引发网络重新进行生成树的计算，生成树的结构将发生相应的变化。不过重新计算得到的新配置消息无法立刻传遍整个网络，如果新选出的根端口和指定端口立刻就开始数据转发的话，可能会造成临时环路。为此，STP 采用了一种端口状态迁移机制，新选出的根端口和指定端口要经过 2 倍的 Forward Delay 延时后才能进入转发状态，这个延时保证了新的配置消息传遍整个网络，使所有参与 STP 计算的交换都能正确知晓网络状态，从而防止了临时环路的产生。在华为交换机设备上，由于默认生成树模式为 MSTP，当手工更改生成树模式为 STP 时，STP 的端口状态同样只有 Discarding、Learning、Forwarding 3 种。在根交换机上配置的延迟时间将作为整个生成树内所有交换机的延迟时间。

超时时间＝3×Hello Time×Timer Factor。如果交换机在配置的超时时间内没有收到上游发送的 BPDU，就认为上游交换机已经出现故障，然后会重新进行生成树拓扑的计算。但是有时交换机在较长的时间内收不到上游发送的 BPDU，是由于上游交换机的繁忙造成的，在这种情况下一般不应该重新进行生成树计算。因此，在稳定的网络中，应将超时时间配置得长一些，以减少网络资源的浪费。建议将 Timer Factor 的值设置为 5～7，以增强网络稳定性。

根交换机的 Hello Time、Forward Delay 以及 Max Age 3 个时间参数之间取值应该满足如下公式，否则网络会频繁震荡。

$$2\times(\text{Forward Delay}-1.0\ \text{second})\geq \text{Max Age}$$

$$\text{Max Age}\geq 2\times(\text{Hello Time}+1.0\ \text{second})$$

建议使用 **stp bridge-diameter** 命令配置网络直径，交换机会自动根据网络直径计算出 Hello Time、Forward Delay 以及 Max Age 3 个时间参数的最优值。默认网络直径为 7。

实验目的

- 理解 STP 中定时器的作用
- 掌握 STP 定时器的配置命令
- 掌握查看 STP 定时器的生效方法
- 理解 STP 定时器的最佳设置方法

实验内容

本实验模拟企业网络场景。公司内网是一个大的局域网，由 4 台交换机两两相连组成的一个环形网络。为了避免形成环路，每台交换机都运行了 STP 生成树协议，且配置 S1 为根交换机，S2 为备份根交换机。现在为了优化网络，在网络变化时加快 STP 的收敛速度，需要在交换机上更改 STP 定时器的设置，将所有定时器调整到最优值，完成 STP 的加速收敛。

实验拓扑

配置 STP 定时器的拓扑如图 4-2 所示。

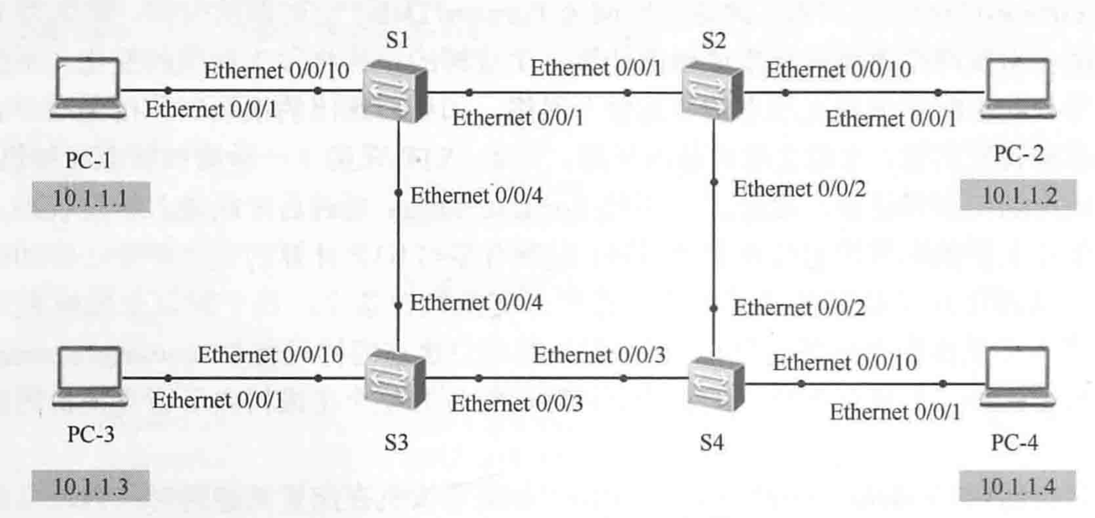


图 4-2　配置 STP 定时器拓扑

实验编址

实验编址见表 4-2。

表 4-2　　实验编址

设备	接口	IP 地址	子网掩码	网关
PC-1	Ethernet 0/0/1	10.1.1.1	255.255.255.0	N/A
PC-2	Ethernet 0/0/1	10.1.1.2	255.255.255.0	N/A
PC-3	Ethernet 0/0/1	10.1.1.3	255.255.255.0	N/A
PC-4	Ethernet 0/0/1	10.1.1.4	255.255.255.0	N/A

MAC 地址

本实验的 MAC 地址见表 4-3。

表 4-3　　**MAC 地址**

设备	全局 MAC 地址
S1（S3700）	4c1f-cc73-c72d
S2（S3700）	4c1f-ccf4-3d05
S3（S3700）	4c1f-cca5-443e
S4（S3700）	4c1f-ccb9-8907

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 地址配置，并使用 **ping** 命令检测各直连链路的连通性。

```
PC>ping 10.1.1.2
Ping 10.1.1.2: 32 data bytes, Press Ctrl_C to break
From 10.1.1.2: bytes=32 seq=1 ttl=128 time=47 ms
From 10.1.1.2: bytes=32 seq=2 ttl=128 time=32 ms
From 10.1.1.2: bytes=32 seq=3 ttl=128 e=78 ms
From 10.1.1.2: bytes=32 seq=4 ttl=128 time=79 ms
From 10.1.1.2: bytes=32 seq=5 ttl=128 time=46 ms
--- 10.1.1.2 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 32/56/79 ms
```

其余直连网段的连通性测试省略。

2. 配置 STP 定时器

在 4 台交换机上配置使用 STP，并配置 S1 为该二层网络中的根交换机，S2 为备份根交换机。

```
[S1]stp enable
[S1]stp mode stp
[S1]stp root primary

[S2]stp enable
[S2]stp mode stp
[S2]stp root secondary

[S3]stp enable
[S3]stp mode stp

[S4]stp enable
[S4]stp mode stp
```

配置完成后，使用 **display stp** 命令查看各定时器的默认值。

```
<S1>display stp
-------[CIST Global Info][Mode STP]-------
CIST Bridge          :0      .4c1f-cc73-c72d
Config Times         :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
Active Times         :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
CIST Root/ERPC       :0      .4c1f-cc73-c72d / 0
……
```

可以查看到在默认情况下，BPDU 每 2 秒发送一次（Hello），BPDU 的最大老化时间为 20s（MaxAge），转发延迟为 15s（FwDly），最大传递跳数为 20 跳（MaxHop）。注意，Config Times 标识的是当前设备配置的计时器，而 Active Times 标识的是正在生效的计时器，一般情况下二者是完全相同的。

在 PC-4 上使用 **ping -t** 命令持续发送 ICMP 报文，进行连通性测试。

```
PC>ping 10.1.1.2 -t
Ping 10.1.1.2: 32 data bytes, Press Ctrl_C to break
From 10.1.1.2: bytes=32 seq=1 ttl=128 time=109 ms
From 10.1.1.2: bytes=32 seq=2 ttl=128 time=109 ms
From 10.1.1.2: bytes=32 seq=3 ttl=128 time=79 ms
From 10.1.1.2: bytes=32 seq=4 ttl=128 time=78 ms
From 10.1.1.2: bytes=32 seq=5 ttl=128 time=109 ms
……
```

可以观察到，此时网络稳定，没有出现任何丢包现象。

在S1上修改STP的Forward Delay时间为2000cs，默认为1500cs，cs代表百分之一秒。注意，只有在根交换机上进行该配置才会生效。

```
[S1]stp timer forward-delay 2000
```

配置完成后，交换机会弹出信息，提示配置已经被改变。

```
[S1]Jun 21 2013 05:57:28-08:00 S1 DS/4/DATASYNC_CFGCHANGE:OID 1.3.6.1.4.1.2011.5.25.191.3.1 configurations have been changed.
```

使用 **display stp** 命令查看此时的定时器值。

```
<S1>display stp
-------[CIST Global Info][Mode STP]-------
CIST Bridge         :0      .4c1f-cc73-c72d
Config Times        :Hello 2s MaxAge 20s FwDly 20s MaxHop 20
Active Times        :Hello 2s MaxAge 20s FwDly 20s MaxHop 20
CIST Root/ERPC      :0      .4c1f-cc73-c72d / 0
......
```

可以观察到，此时修改已经完成。如果在非根交换机上配置，那么Config Times配置值会发生改变，而Active Times实际运行值不会改变。

再回到PC-4上观察到PC-2的连通性测试结果。

```
From 10.1.1.2: bytes=32 seq=46 ttl=128 time=78 ms
From 10.1.1.2: bytes=32 seq=47 ttl=128 time=109 ms
From 10.1.1.2: bytes=32 seq=48 ttl=128 time=110 ms
From 10.1.1.2: bytes=32 seq=49 ttl=128 time=78 ms
Request timeout!
Request timeout!
Request timeout!
......
```

观察到出现大量丢包现象。

如果更改STP的Hello Time时间及其他计时器也会出现相同的现象，这里不再赘述。所以不建议使用命令直接修改定时器时间，而建议使用**stp bridge-diameter**命令设置网络直径，交换机会根据网络直径自动计算出3个时间参数的最优值。注意，本命令需要在根交换机上配置才能生效。

在S1上使用 **stp bridge-diameter 3** 命令设置网络的直径为3。

```
[S1]stp bridge-diameter 3
```

配置完成后，观察STP计时器的改变情况。

```
[S1]display stp
-------[CIST Global Info][Mode STP]-------
CIST Bridge         :0      .4c1f-cc73-c72d
Config Times        :Hello 2s MaxAge 12s FwDly 9s MaxHop 20
Active Times        :Hello 2s MaxAge 12s FwDly 9s MaxHop 20
......
```

可以观察到，此时最大老化时间被自动修改为12s，转发延迟被自动修改为9s。

同时对PC-4到PC-2连通性测试结果再次进行观察。

```
From 10.1.1.2: bytes=32 seq=18 ttl=128 time=63 ms
From 10.1.1.2: bytes=32 seq=19 ttl=128 time=78 ms
Request timeout!
Request timeout!
......
Request timeout!
```

```
Request timeout!
From 10.1.1.2: bytes=32 seq=30 ttl=128 time=78 ms
From 10.1.1.2: bytes=32 seq=31 ttl=128 time=78 ms
From 10.1.1.2: bytes=32 seq=32 ttl=128 time=62 ms
```

可以观察到，此时网络恢复了正常。

3. 验证 Forward Delay 定时器

为了验证 Forward Delay 时间对端口状态迁移的影响，仍然维持上一步骤中 PC-4 到 PC-2 的连通性测试。

在 S1、S2、S3、S4 上查看 STP 下的各个端口的状态。

```
<S1>display stp brief
 MSTID  Port                    Role   STP State     Protection
   0    Ethernet0/0/1           DESI   FORWARDING      NONE
   0    Ethernet0/0/4           DESI   FORWARDING      NONE
   0    Ethernet0/0/10          DESI   FORWARDING      NONE
```

可以观察到，由于 S1 是根交换机，所以 S1 的所有端口都是 DP 端口即指定端口，所处状态都是转发状态。

同理观察其他交换机的 STP 接口状态。

```
<S2>display stp brief
 MSTID  Port                    Role   STP State     Protection
   0    Ethernet0/0/1           ROOT   FORWARDING      NONE
   0    Ethernet0/0/2           DESI   FORWARDING      NONE
   0    Ethernet0/0/10          DESI   FORWARDING      NONE

<S3>display stp brief
 MSTID  Port                    Role   STP State     Protection
   0    Ethernet0/0/3           DESI   FORWARDING      NONE
   0    Ethernet0/0/4           ROOT   FORWARDING      NONE
   0    Ethernet0/0/10          DESI   FORWARDING      NONE

<S4>display stp brief
 MSTID  Port                    Role   STP State     Protection
   0    Ethernet0/0/2           ROOT   FORWARDING      NONE
   0    Ethernet0/0/3           ALTE   DISCARDING      NONE
   0    Ethernet0/0/10          DESI   FORWARDING      NONE
```

可以观察到，此时 S2 和 S3 接口都属于转发状态，S4 的 E 0/0/2 接口为根端口，E 0/0/3 接口处于阻塞状态。

现在将 S4 的 E 0/0/2 接口关闭，使 E 0/0/3 接口成为新的根端口。

请注意，在华为交换机上，当从 MSTP 模式切换到 STP 模式，运行 STP 协议的设备上端口支持的端口状态仍然保持和 MSTP 支持的端口状态一样，仅包括 Forwarding、Learning 和 Discarding。又由于华为交换机上默认的 STP 模式为 MSTP，故本实验中，STP 仅支持 3 个状态，S4 的 E 0/0/3 接口会从 Discarding 状态，再经过 Learning 过渡状态，最终到 Forwarding 状态，只需经历一个 Forward Delay 的时间。

```
[S4]interface Ethernet 0/0/2
[S4-Ethernet0/0/2]shutdown
```

配置完成后，观察连通性测试结果。

```
From 10.1.1.2: bytes=32 seq=11 ttl=128 time=109 ms
From 10.1.1.2: bytes=32 seq=12 ttl=128 time=109 ms
From 10.1.1.2: bytes=32 seq=13 ttl=128 time=47 ms
```

```
From 10.1.1.2: bytes=32 seq=14 ttl=128 time=47 ms
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
From 10.1.1.2: bytes=32 seq=15 ttl=128 time=47 ms
From 10.1.1.2: bytes=32 seq=16 ttl=128 time=63 ms
From 10.1.1.2: bytes=32 seq=17 ttl=128 time=32 ms
```

可以观察到，此时丢失了 9 个数据包。这是因为根据上一步骤的配置结果，Forward Delay 时间为 9s，即 S4 上该端口从 Discarding 状态经过 Learning 状态，最终到 Forwarding 状态需要一个 Forwad Delay 的时间间隔。

恢复 S4 的 E 0/0/2 接口，并在根交换机 R1 上更改网络直径为默认值 7。

```
[S4]interface Ethernet 0/0/2
[S4-Ethernet0/0/2]undo shutdown

[S1]stp bridge-diameter 7
```

配置完成后，查看 S1 上的 STP 信息。

```
[S1]display stp
-------[CIST Global Info][Mode STP]-------
CIST Bridge          :0     .4c1f-cc73-c72d
Config Times         :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
Active Times         :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
......
```

可以观察到，Forward 时间已被自动修改为 15s。

现在采用相同的方法，关闭 S4 上的 E 0/0/2 接口，测试丢包情况。并在配置完成后，观察连通性测试结果。

```
From 10.1.1.2: bytes=32 seq=33 ttl=128 time=53 ms
From 10.1.1.2: bytes=32 seq=34 ttl=128 time=17 ms
From 10.1.1.2: bytes=32 seq=35 ttl=128 time=30 ms
From 10.1.1.2: bytes=32 seq=36 ttl=128 time=45 ms
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
```

```
From 10.1.1.2: bytes=32 seq=37 ttl=128 time=31 ms
From 10.1.1.2: bytes=32 seq=38 ttl=128 time=37 ms
From 10.1.1.2: bytes=32 seq=39 ttl=128 time=52 ms
```

可以观察到，丢包共 17 个，即验证了端口状态迁移从 Discarding 状态到 Forwarding 状态经过了一个 Forward Delay 的 15s 时间间隔。

思考

交换机端口在发生状态转换时，都有哪些状态会经历一个 Forward Delay？

4.3 RSTP 基础配置

原理概述

IEEE 于 2001 年发布的 802.1w 标准定义了 RSTP（Rapid Spanning-Tree Protocol，快速生成树协议），该协议基于 STP 协议，对原有的 STP 协议进行了更加细致的修改和补充。

STP 协议虽然能够解决环路问题，但是也存在一些不足，比如 STP 没有细致区分端口状态和端口角色；其次 STP 端口状态共有 5 种，即 Discarding、Blocking、Listening、Learning 和 Forwarding，收敛较慢。而且，对于用户来说 Listening、Learning 和 Blocking 状态并没有区别，都不转发流量。根据 STP 的不足，RSTP 做出了改进。

RSTP 新增加了 2 种端口角色，其端口角色共有 4 种：根端口、指定端口、Alternate 端口和 Backup 端口。根端口和指定端口的作用与 STP 协议中相同，Alternate 端口和 Backup 端口的描述如下：

■ Alternate 端口就是由于学习（Learning）到其他网桥发送的配置 BPDU 报文而阻塞的端口，Alternate 端口提供了从指定桥到根的另一条可切换路径，作为根端口的备份端口；

■ Backup 端口就是由于学习到自身发送的配置 BPDU 报文而阻塞的端口，Backup 端口作为指定端口的备份，提供了另一条从根桥到相应网段的备份通路。

RSTP 把原来的 5 种状态缩减为 3 种。根据端口是否转发用户流量和学习 MAC 地址来划分：如果不转发用户流量也不学习 MAC 地址，那么端口状态就是 Discarding 状态；如果不转发用户流量但是学习 MAC 地址，那么端口状态就是 Learning 状态；如果既转发用户流量又学习 MAC 地址，那么端口状态就是 Forwarding 状态。

RSTP 的快速收敛机制可分为以下 3 种。

■ Proposal/Agreement 机制：当一个端口被选举成为指定端口之后，在 STP 中，该端口至少要等待一个 Forward Delay（Learning）时间才会迁移到 Forwarding 状态。而在 RSTP 中，此端口会先进入 Discarding 状态，再通过 Proposal/Agreement 机制（可简称“P/A 机制”）快速进入 Forwarding 状态。这种机制必须在点到点全双工链路上使用；

■ 根端口快速切换机制：如果网络中一个根端口失效，那么网络中最优的 Alternate 端口将成为根端口，进入 Forwarding 状态。因为通过这个 Alternate 端口连接的网段上必

然有个指定端口可以通往根桥。

■ 边缘端口的引入：在 RSTP 里面，如果某一个指定端口位于整个网络的边缘，即不再与其他交换设备连接，而是直接与终端设备直连，这种端口叫做边缘端口。边缘端口不接收处理配置 BPDU，不参与 RSTP 运算，可以由 Disable 直接转到 Forwarding 状态，且不经历时延，就像在端口上将 STP 禁用。但是一旦边缘端口收到配置 BPDU，就丧失了边缘端口属性，成为普通 STP 端口，并重新进行生成树计算，从而引起网络震荡。

实验目的

- 理解 RSTP 的应用场景
- 掌握 RSTP 的基本配置
- 掌握 RSTP 的边缘端口的应用
- 理解 RSTP 备份端口

实验内容

本实验模拟公司网络场景。S3 和 S4 是接入层交换机，负责用户的接入，S1 和 S2 是汇聚层交换机，四台交换机组成一个环形网络。为了防止网络中出现环路，产生网络风暴，所有交换机上都需要运行生成树协议。同时为了加快网络收敛速度，网络管理员选择使用 RSTP 协议，且使得性能较好的 S1 为根交换机，S2 为次根交换机，并配置边缘端口进一步优化公司网络。

实验拓扑

RSTP 基础配置拓扑如图 4-3 所示。

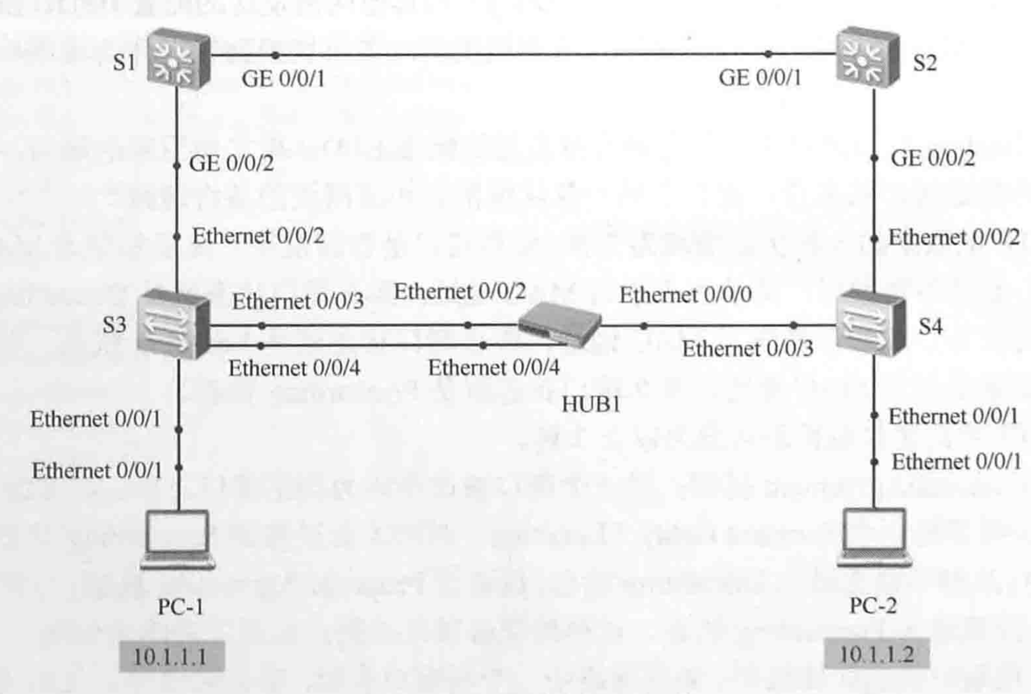


图 4-3 RSTP 基础配置拓扑

实验编址

实验编址见表 4-4。

表 4-4　实验编址

设备	接口	IP 地址	子网掩码	默认网关
PC-1	Ethernet 0/0/1	10.1.1.1	255.255.255.0	N/A
PC-2	Ethernet 0/0/1	10.1.1.2	255.255.255.0	N/A

MAC 地址

本实验的 MAC 地址见表 4-5。

表 4-5　**MAC** 地址

设备	全局 MAC 地址
S1（S3700）	4c1f-cc33-9812
S2（S3700）	4c1f-cc4a-bea9
S3（S3700）	4c1f-cc32-b9d7
S4（S3700）	4c1f-cc10-279a

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 地址配置，并使用 **ping** 命令检测各直连链路的连通性。PC-1 的配置如图 4-4 所示，PC-2 的配置方法相同，此处省略。

图 4-4　PC-1 配置界面

配置完成后，测试主机间的连通性。

```
PC>ping 10.1.1.2
Ping 10.1.1.2: 32 data bytes, Press Ctrl_C to break
From 10.1.1.2: bytes=32 seq=1 ttl=128 time=47 ms
From 10.1.1.2: bytes=32 seq=2 ttl=128 time=62 ms
From 10.1.1.2: bytes=32 seq=3 ttl=128 time=32 ms
From 10.1.1.2: bytes=32 seq=4 ttl=128 time=78 ms
From 10.1.1.2: bytes=32 seq=5 ttl=128 time=31 ms
--- 10.1.1.2 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 31/50/78 ms
```

可以观察到，连通性测试成功。

2. 配置 RSTP 基本功能

在汇聚层交换机 S1、S2 及接入层交换机 S3、S4 上，把生成树模式由默认的 MSTP 改为 RSTP。由于华为交换机上默认即开启了 MSTP，故只需修改生成树模式即可。

```
[S1]stp mode rstp
Info: This operation may take a few seconds. Please wait for a moment...done.

[S2]stp mode rstp
Info: This operation may take a few seconds. Please wait for a moment...done.

[S3]stp mode rstp
Info: This operation may take a few seconds. Please wait for a moment...done.

[S4]stp mode rstp
Info: This operation may take a few seconds. Please wait for a moment...done.
```

配置完成后，在交换机 S1、S2、S3 和 S4 上都使用 **display stp** 命令去查看生成树的模式及根交换机的位置。

```
<S1>display stp
-------[CIST Global Info][Mode RSTP]-------
CIST Bridge          :32768.4c1f-cc33-9812
Config Times         :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
Active Times         :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
CIST Root/ERPC       :32768.4c1f-cc10-279a / 2
CIST RegRoot/IRPC    :32768.4c1f-cc33-9812 / 0
......

<S2>display stp
-------[CIST Global Info][Mode RSTP]-------
CIST Bridge          :32768.4c1f-cc4a-bea9
Config Times         :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
Active Times         :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
CIST Root/ERPC       :32768.4c1f-cc10-279a / 1
CIST RegRoot/IRPC    :32768.4c1f-cc4a-bea9 / 0
......

<S3>display stp
-------[CIST Global Info][Mode RSTP]-------
CIST Bridge          :32768.4c1f-cc32-b9d7
Config Times         :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
```

```
Active Times          :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
CIST Root/ERPC        :32768.4c1f-cc10-279a / 1
CIST RegRoot/IRPC     :32768.4c1f-cc32-b9d7 / 0
……

<S4>display stp
-------[CIST Global Info][Mode RSTP]-------
CIST Bridge           :32768.4c1f-cc10-279a
Config Times          :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
Active Times          :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
CIST Root/ERPC        :32768.4c1f-cc10-279a / 0
CIST RegRoot/IRPC     :32768.4c1f-cc10-279a / 0
……
```

上述信息中，CIST Bridge 是交换机自己的 ID，而 CIST Root 是根交换机的 ID。根交换机是交换机 ID 最小的交换机，所以，观察可知，S4 是当前的根交换机。

在 RSTP 构建的树形拓扑中，网络管理员需要设置汇聚层主交换机 S1 为根交换机，汇聚层交换机 S2 为备份根交换机。

```
[S1]stp root primary

[S2]stp root secondary
```

配置完成后，同样在 S1 上使用 **display stp** 命令观察。

```
[S1]display stp
-------[CIST Global Info][Mode RSTP]-------
CIST Bridge           :0     .4c1f-cc33-9812
Config Times          :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
Active Times          :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
CIST Root/ERPC        :0     .4c1f-cc33-9812 / 0
CIST RegRoot/IRPC     :0     .4c1f-cc33-9812 / 0
CIST RootPortId       :0.0
BPDU-Protection       :Disabled
CIST Root Type        :Primary root
TC or TCN received    :33
……
```

可以观察到，**stp root primary** 命令修改的是交换机 ID 中的交换机优先级，把默认的优先级由 32768 改为 0，所以 S1 的交换机 ID 变为最小，是 Primary root，即根交换机。

在 S2 上使用 **display stp** 命令观察。

```
[S2]display stp
-------[CIST Global Info][Mode RSTP]-------
CIST Bridge           :4096 .4c1f-cc4a-bea9
Config Times          :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
Active Times          :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
CIST Root/ERPC        :0     .4c1f-cc33-9812 / 1
CIST RegRoot/IRPC     :4096 .4c1f-cc4a-bea9 / 0
CIST RootPortId       :128.1
BPDU-Protection       :Disabled
CIST Root Type        :Secondary root
TC or TCN received    :23
……
```

可以观察到，**stp root secondary** 命令修改的也是交换机 ID 中的交换机优先级，把默认的优先级由 32768 改为 4096，使 S2 的桥 ID 变为次小，是 Secondary root，即次根交换机。

在 S3 和 S4 上使用 **display stp** 命令观察。

```
<S3>display stp
-------[CIST Global Info][Mode RSTP]-------
CIST Bridge          :32768.4c1f-cc32-b9d7
Config Times         :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
Active Times         :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
CIST Root/ERPC       :0     .4c1f-cc33-9812 / 1
CIST RegRoot/IRPC    :32768.4c1f-cc32-b9d7 / 0
……

[S4]display stp
-------[CIST Global Info][Mode RSTP]-------
CIST Bridge          :32768.4c1f-cc10-279a
Config Times         :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
Active Times         :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
CIST Root/ERPC       :0     .4c1f-cc33-9812 / 2
CIST RegRoot/IRPC    :32768.4c1f-cc10-279a / 0
……
```

可以观察到，S3 和 S4 交换机的交换机优先级保持默认的 32768，且都把 S1 当作根交换机。

继续使用 **display stp brief** 命令查看每台交换机上的端口角色及状态。

```
<S1>display stp brief
 MSTID  Port                        Role  STP State     Protection
   0    GigabitEthernet0/0/1        DESI  FORWARDING      NONE
   0    GigabitEthernet0/0/2        DESI  FORWARDING      NONE
```

根交换机 S1 上无根端口，所有端口都是指定端口。

```
<S2>display stp brief
 MSTID  Port                        Role  STP State     Protection
   0    GigabitEthernet0/0/1        ROOT  FORWARDING      NONE
   0    GigabitEthernet0/0/2        DESI  FORWARDING      NONE
```

交换机 S2 上的 GE 0/0/1 是根端口。

```
<S3>display stp brief
 MSTID  Port                        Role  STP State     Protection
   0    Ethernet0/0/1               DESI  FORWARDING      NONE
   0    Ethernet0/0/2               ROOT  FORWARDING      NONE
   0    Ethernet0/0/3               DESI  FORWARDING      NONE
   0    Ethernet0/0/4               BACK  DISCARDING      NONE
```

交换机 S3 上的 E 0/0/2 是根端口，E 0/0/3 是指定端口，而 E 0/0/4 是备份端口。

```
<S4>display stp brief
 MSTID  Port                        Role  STP State     Protection
   0    Ethernet0/0/1               DESI  FORWARDING      NONE
   0    Ethernet0/0/2               ROOT  FORWARDING      NONE
   0    Ethernet0/0/3               ALTE  DISCARDING      NONE
```

交换机 S4 上的 E 0/0/2 是根端口，E 0/0/3 是替代端口。

通过下面的操作，观察 S2 上端口的状态变化。

目前 S2 的 GE 0/0/1 端口是根端口，其他所有端口是指定端口。如果 S2 的根端口断掉了，S2 会选择把其他到达根交换机的端口置成根端口。RSTP 协议的收敛比较快，端口 GE 0/0/2 会快速协商成为新的根端口，协商期间端口是 Discarding 状态，协商结束后端口为 Forwarding 状态，这个过程所需要的时间非常短，这就是 RSTP 收敛快的一个表现。

模拟根端口断掉的过程，把 S2 的 GE 0/0/1 端口使用 **shutdown** 关闭，同时，使用 **display stp brief** 命令观察 S2 上其他端口的角色及状态的变化。

```
[S2]interface GigabitEthernet0/0/1
[S2-GigabitEthernet0/0/1]shutdown
Jun 26 2013 01:01:24-08:00 S2 %%01PHY/1/PHY(l)[2]:      GigabitEthernet0/0/1: chang
e status to down

[S2-GigabitEthernet0/0/1]display stp brief
 MSTID  Port                        Role  STP State     Protection
   0    GigabitEthernet0/0/2        DESI  DISCARDING      NONE
```

可以观察到，端口 GE 0/0/2 的角色还是指定端口，但状态是 Discarding。再次使用 **display stp brief** 命令时，就会观察到端口的角色为根端口，且处于转发状态。

```
[S2-GigabitEthernet0/0/1]display stp brief
 MSTID  Port                        Role  STP State     Protection
   0    GigabitEthernet0/0/2        ROOT  FORWARDING      NONE
```

观察结束之后，恢复端口。

```
[S2-GigabitEthernet0/0/1]undo shutdown
Jun 26 2013 01:01:47-08:00 S2 %%01PHY/1/PHY(l)[4]:      GigabitEthernet0/0/1: chang
e statu
[S2-GigabitEthernet0/0/1]display stp brief
 MSTID  Port                        Role  STP State     Protection
   0    GigabitEthernet0/0/1        ROOT  FORWARDING      NONE
   0    GigabitEthernet0/0/2        DESI  DISCARDING      NONE
```

可以观察到，端口 GE 0/0/2 的角色是指定端口，状态是 Discarding。再次使用 **display stp brief** 命令时，就会观察到 GE 0/0/2 会经历 Discarding 状态回到 Forwarding 状态。

```
[S2-GigabitEthernet0/0/1]display stp brief
 MSTID  Port                        Role  STP State     Protection
   0    GigabitEthernet0/0/1        ROOT  FORWARDING      NONE
   0    GigabitEthernet0/0/2        DESI  FORWARDING      NONE
```

当拓扑发生变化时，RSTP 使用 P/A 机制和根端口快速切换机制使端口状态立即从 Discarding 进入 Forwarding 状态，缩短了收敛的时间，减小了对网络通信的影响。

3. 配置边缘端口

生成树的计算主要发生在交换机互连的链路之上，而连接 PC 的端口没有必要参与生成树计算，为了优化网络，降低生成树计算对终端设备的影响，现在网络管理员把交换机上连接 PC 的接口配置为边缘端口。

作为对比，在将 S4 上的 E 0/0/1 配置为边缘端口之前，先把端口关闭再开启，观察端口状态的变化。

```
[S4]display stp brief
 MSTID  Port                        Role  STP State     Protection
   0    Ethernet0/0/1               DESI  FORWARDING      NONE
   0    Ethernet0/0/2               ROOT  FORWARDING      NONE
   0    Ethernet0/0/3               ALTE  DISCARDING      NONE
[S4]interface Ethernet0/0/1
[S4-Ethernet0/0/1]shutdown
[S4-Ethernet0/0/1]undo shutdown
[S4-Ethernet0/0/1]display stp brief
 MSTID  Port                        Role  STP State     Protection
   0    Ethernet0/0/1               DESI  DISCARDING      NONE
   0    Ethernet0/0/2               ROOT  FORWARDING      NONE
   0    Ethernet0/0/3               ALTE  DISCARDING      NONE
```

可以观察到初始状态为 Discarding，15 秒之后，接口将进入 Learning 状态。

```
[S4-Ethernet0/0/1]display stp brief
 MSTID  Port                        Role   STP State    Protection
   0    Ethernet0/0/1               DESI   LEARNING       NONE
   0    Ethernet0/0/2               ROOT   FORWARDING     NONE
   0    Ethernet0/0/3               ALTE   DISCARDING     NONE
```

保持在 Learning 状态 15s 后，接口最终进入到 Forwarding 状态。

```
[S4-Ethernet0/0/1]display stp brief
MSTID  Port                         Role   STP State    Protection
   0    Ethernet0/0/1               DESI   FORWARDING     NONE
   0    Ethernet0/0/2               ROOT   FORWARDING     NONE
   0    Ethernet0/0/3               ALTE   DISCARDING     NONE
```

所以一个接口如果参与生成树计算，要经过 Discarding 和 Learning 状态，30s 后才最终进入转发状态。

配置 S4 上连接 PC 的端口为边缘端口，此时生成树计算工作依然进行，但端口进入转发状态无需等待 30s。

```
[S4]interface Ethernet0/0/1
[S4-Ethernet0/0/1]stp edged-port enable
```

在 S4 上，做同样的模拟过程，关闭 E 0/0/1 接口，再重新开启此端口，观察边缘端口 E 0/0/1 的状态变化。

```
[S4-Ethernet0/0/1]shutdown
[S4-Ethernet0/0/1]undo shutdown
[S4-Ethernet0/0/1]display stp brief
 MSTID  Port                        Role   STP State    Protection
   0    Ethernet0/0/1               DESI   FORWARDING     NONE
   0    Ethernet0/0/2               ROOT   FORWARDING     NONE
   0    Ethernet0/0/3               ALTE   DISCARDING     NONE
```

可以观察到，接口立刻进入到 Forwarding 状态，没有 30s 的延迟。

在使用 RSTP 的环境中，可以在交换机上把连接 PC、路由器和防火墙的端口都配置为边缘端口，边缘端口能降低终端设备访问网络需要等待的时间，明显提高网络的可用性。

4. 查看备份端口状态

网络管理员在 S3 与 S4 之间加了一台 Hub 设备，并将 S3 的 E 0/0/4 通过 Hub 与 S4 相连。

在 S3 上使用 **display stp brief** 命令查看生成树信息。

```
[S3]display stp brief
 MSTID  Port                        Role   STP State    Protection
   0    Ethernet0/0/1               DESI   FORWARDING     NONE
   0    Ethernet0/0/2               ROOT   FORWARDING     NONE
   0    Ethernet0/0/3               DESI   FORWARDING     NONE
   0    Ethernet0/0/4               BACK   DISCARDING     NONE
```

可以观察到，S3 的 E 0/0/3 接口为指定端口，而同交换机上的 E 0/0/4 为备份端口，两个接口接到同一台 Hub 上，当 E 0/0/3 接口关闭之后，E 0/0/4 会成为新的指定端口。

在 S3 上关闭 E 0/0/3 接口，通过 **display stp brief** 命令查看备份端口的状态变化。

```
[S3]interface Ethernet0/0/3
[S3-Ethernet0/0/3]shutdown
```

```
[S3-Ethernet0/0/3]display stp brief
 MSTID  Port                    Role   STP State     Protection
   0    Ethernet0/0/1           DESI   FORWARDING      NONE
   0    Ethernet0/0/2           ROOT   FORWARDING      NONE
   0    Ethernet0/0/4           BACK   DISCARDING      NONE

[S3-Ethernet0/0/3]display stp brief
 MSTID  Port                    Role   STP State     Protection
   0    Ethernet0/0/1           DESI   FORWARDING      NONE
   0    Ethernet0/0/2           ROOT   FORWARDING      NONE
   0    Ethernet0/0/4           DESI   LEARNING        NONE

[S3-Ethernet0/0/3]display stp brief
 MSTID  Port                    Role   STP State     Protection
   0    Ethernet0/0/1           DESI   FORWARDING      NONE
   0    Ethernet0/0/2           ROOT   FORWARDING      NONE
   0    Ethernet0/0/4           DESI   FORWARDING      NONE
[S3-Ethernet0/0/3]
```

可以观察到，S3 上的指定接口断掉后，E 0/0/4 接口角色发生变化，状态会由 Discarding、Learning 最终到 Forwarding 状态，指定接口现在是 E 0/0/4，指定交换机还是 S3，S3 仍然为 Hub 所在的网段提供访问其他交换机的数据访问路径。

相似的过程，在 S4 上，接口 E 0/0/2 是根端口，接口 E 0/0/3 是替代端口，Discarding 状态。当 S4 的根端口 E 0/0/2 关闭之后，接口 E 0/0/3 会立即替代 E 0/0/2 成为新的根端口。

```
[S4]display stp brief
 MSTID  Port                    Role   STP State     Protection
   0    Ethernet0/0/1           DESI   FORWARDING      NONE
   0    Ethernet0/0/2           ROOT   FORWARDING      NONE
   0    Ethernet0/0/3           ALTE   DISCARDING      NONE
```

把 S4 上的根端口 E 0/0/2 关闭掉，观察替代端口 E 0/0/3 的状态及角色的变化。

```
[S4]interface ethernet0/0/2
[S4-Ethernet0/0/2]shutdown
[S4-Ethernet0/0/2]display stp brief
 MSTID  Port                    Role   STP State     Protection
   0    Ethernet0/0/1           DESI   FORWARDING      NONE
   0    Ethernet0/0/3           ROOT   FORWARDING      NONE
```

RSTP 协议收敛很快，所以替代端口立即成为根端口。

在 RSTP 中，Alternate 端口和 Backup 端口角色所对应的最终端口状态都是 Discarding。区别是 Alternate 端口用于为根端口做备份，而 Backup 端口用于为本交换机上的指定端口做备份，所以当相应的根端口或指定端口断掉后，备份端口会立即承担原有的根端口或指定端口的角色，开始转发数据。

RSTP 协议是对 STP 的升级，它重新划定端口的角色及状态，使用更快速的握手协商机制，降低了收敛时间，使它成为继 STP 协议后首选的生成树协议，不足之处就是在同一网络内的交换机上所有的 VLAN 共用同样的拓扑，此时可以使用 MSTP 来优化。

思考

S4 交换机的 E 0/0/2 接口关闭之后，E 0/0/3 会成为新的根端口，如果此时 S3 交换机

的指定端口 E 0/0/3 也关闭掉，S4 交换机上会发生端口角色或状态的改变吗？如果边缘端口收到 BPDU，此端口还是边缘端口吗？

4.4 MSTP 基础配置

原理概述

RSTP 在 STP 基础上进行了改进，实现了网络拓扑快速收敛。但 RSTP 和 STP 还存在同一个缺陷，即由于局域网内所有的 VLAN 共享一棵生成树，链路被阻塞后将不承载任何流量，造成带宽浪费，因此无法在 VLAN 间实现数据流量的负载均衡，还有可能造成部分 VLAN 的报文无法转发。

通过 MSTP 把一个交换网络划分成多个域，每个域内形成多棵生成树，生成树之间彼此独立。每个域叫做一个 MST 域（Multiple Spanning Tree Region，MST Region），每棵生成树叫做一个多生成树实例 MSTI（Multiple Spanning Tree Instance）。

实例内可以包含多个 VLAN。通过将多个 VLAN 映射到同一个实例内，可以节省通信开销和资源占用率。MSTP 各个实例拓扑的生成树计算相互独立，通过这些实例可以实现负载均衡。把多个相同拓扑结构的 VLAN 映射到一个实例里，这些 VLAN 在端口上的转发状态取决于端口在对应 MSTP 实例的状态。

MSTP 通过设置 VLAN 映射表（即 VLAN 和 MSTI 的对应关系表），把 VLAN 和 MSTI 联系起来。每个 VLAN 只能对应一个 MSTI，即同一 VLAN 的数据只能在一个 MSTI 中传输，而一个 MSTI 可能对应多个 VLAN。

实验目的

- 掌握 MSTP 的基础配置
- 掌握配置 MSTP 多实例的方法
- 掌握配置 MSTP 实现流量分担的方法
- 理解 MSTP 与 STP、RSTP 的区别

实验内容

某公司二层网络由三台交换机 S1、S2、S3 组成。交换机 S1 与 S2 在一个楼层，S3 在另一楼层。PC-1 与 PC-2 属于 HR 部门，划入 VLAN 10，PC-3 与 PC-4 属于 IT 部门，划入 VLAN 20。当使用普通 STP 时，STP 将会阻塞一条链路来防止环路产生，导致该链路闲置。为了保证所有链路都能充分利用，使流量能够分担，网络管理员通过配置 MSTP 来实现。

实验拓扑

MSTP 基础配置拓扑如图 4-5 所示。

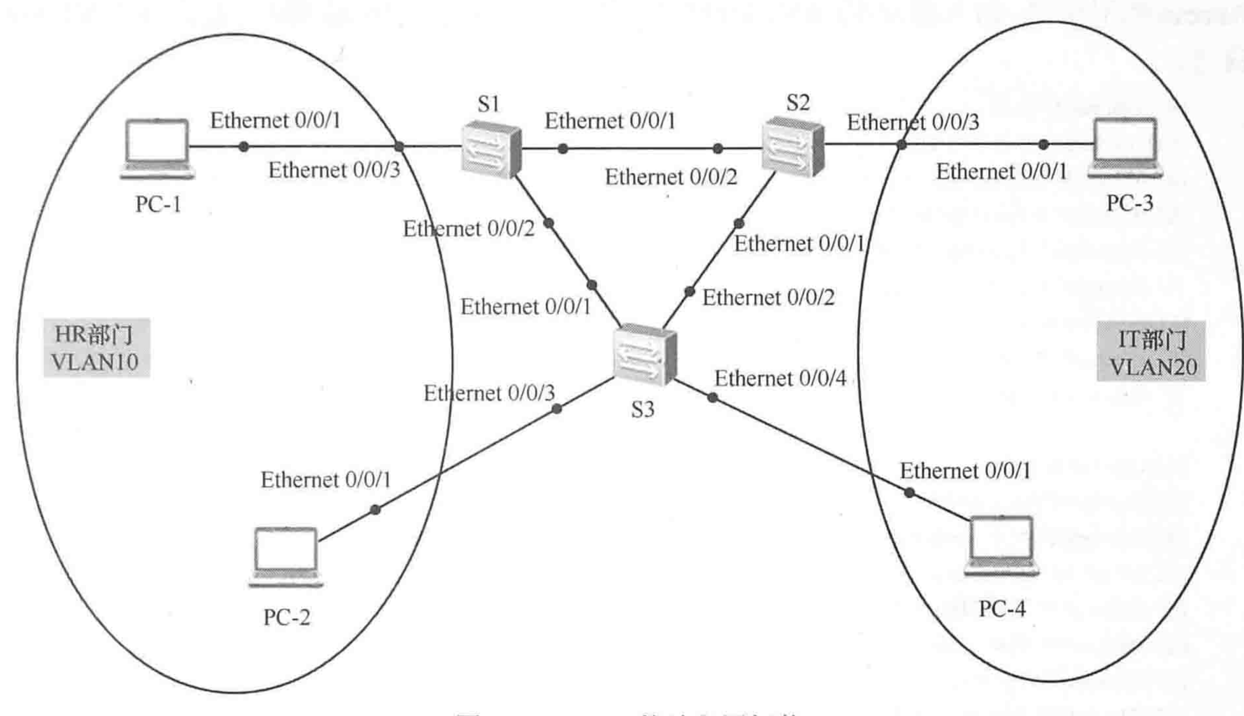


图 4-5　MSTP 基础配置拓扑

实验编址

实验编址见表 4-6。

表 4-6　　实验编址

设备	接口	IP 地址	子网掩码	默认网关
PC-1	Ethernet 0/0/1	192.168.10.1	255.255.255.0	N/A
PC-2	Ethernet 0/0/1	192.168.10.2	255.255.255.0	N/A
PC-3	Ethernet 0/0/1	192.168.20.1	255.255.255.0	N/A
PC-4	Ethernet 0/0/1	192.168.20.2	255.255.255.0	N/A

MAC 地址

本实验的 MAC 地址见表 4-7。

表 4-7　　**MAC 地址**

设备	全局 MAC 地址
S1（S3700）	4c1f-cc4d-0fcf
S2（S3700）	4c1f-cc5e-3ea9
S3（S3700）	4c1f-ccf8-067c

实验步骤

1．基础配置

根据编址表，在各台 PC 上配置 IP 地址。

在交换机 S1、S2、S3 上创建 VLAN 10 与 VLAN 20，并将连接 PC 的端口配置成为

Access类型接口，划入相应VLAN。交换机间的接口配置成为Trunk接口，允许所有VLAN通过。

```
[S1]vlan batch 10 20
[S1]interface Ethernet0/0/3
[S1-Ethernet0/0/3]port link-type access
[S1-Ethernet0/0/3]port default vlan 10
[S1-Ethernet0/0/3]interface Ethernet0/0/1
[S1-Ethernet0/0/1]port link-type trunk
[S1-Ethernet0/0/1]port trunk allow-pass vlan all
[S1-Ethernet0/0/1]interface Ethernet0/0/2
[S1-Ethernet0/0/2]port trunk allow-pass vlan all

[S2]vlan batch 10 20
[S2]interface Ethernet0/0/3
[S2-Ethernet0/0/3]port link-type access
[S2-Ethernet0/0/3]port default vlan 20
[S2-Ethernet0/0/3]interface Ethernet0/0/2
[S2-Ethernet0/0/2]port link-type trunk
[S2-Ethernet0/0/2]port trunk allow-pass vlan all
[S2-Ethernet0/0/2]interface Ethernet0/0/1
[S2-Ethernet0/0/1]port link-type trunk
[S2-Ethernet0/0/1]port trunk allow-pass vlan all

[S3]vlan batch 10 20
[S3]interface Ethernet0/0/3
[S3-Ethernet0/0/3]port link-type access
[S3-Ethernet0/0/3]port default vlan 10
[S3-Ethernet0/0/3]interface Ethernet0/0/4
[S3-Ethernet0/0/4]port link-type access
[S3-Ethernet0/0/4]port default vlan 20
[S3-Ethernet0/0/4]interface Eth0/0/1
[S3-Ethernet0/0/1]port link-type trunk
[S3-Ethernet0/0/1]port trunk allow-pass vlan all
[S3-Ethernet0/0/1]interface Ethernet0/0/2
[S3-Ethernet0/0/2]port link-type trunk
[S3-Ethernet0/0/2]port trunk allow-pass vlan all
```

2. 理解MSTP的运行机制及验证单实例

当网络管理员按照设计搭建完公司二层网络后，启动设备。在华为交换机上默认即运行MSTP协议。

在S1上使用 **display stp** 命令查看生成树的状态和统计信息。

```
<S1>display stp
-------[CIST Global Info][Mode MSTP]-------
CIST Bridge         :32768.4c1f-cc4d-0fcf
Config Times        :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
Active Times        :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
CIST Root/ERPC      :32768.4c1f-cc4d-0fcf / 0
CIST RegRoot/IRPC   :32768.4c1f-cc4d-0fcf / 0
CIST RootPortId     :0.0
BPDU-Protection     :Disabled
TC or TCN received  :5
```

```
TC count per hello    :0
STP Converge Mode     :Normal
Time since last TC    :0 days 0h:24m:48s
Number of TC          :6
Last TC occurred      :Ethernet0/0/1
----[Port1(Ethernet0/0/1)][FORWARDING]----
 Port Protocol        :Enabled
……
```

可以观察到，在 CIST 全局信息中，显示目前 STP 模式为 MSTP，根交换机为 S1 自身，另外还有交换机各个接口上的 STP 信息。

使用 **display stp brief** 命令查看 S1、S2、S3 上生成树的状态和统计的摘要信息。

```
<S1>display stp brief
 MSTID  Port                 Role   STP State     Protection
   0    Ethernet0/0/1        DESI   FORWARDING      NONE
   0    Ethernet0/0/2        DESI   FORWARDING      NONE
   0    Ethernet0/0/3        DESI   FORWARDING      NONE

<S2>display stp brief
 MSTID  Port                 Role   STP State     Protection
   0    Ethernet0/0/1        DESI   FORWARDING      NONE
   0    Ethernet0/0/2        ROOT   FORWARDING      NONE
   0    Ethernet0/0/3        DESI   FORWARDING      NONE

<S3>display stp brief
 MSTID  Port                 Role   STP State     Protection
   0    Ethernet0/0/1        ROOT   FORWARDING      NONE
   0    Ethernet0/0/2        ALTE   DISCARDING      NONE
   0    Ethernet0/0/3        DESI   FORWARDING      NONE
   0    Ethernet0/0/4        DESI   FORWARDING      NONE
```

可以观察到，此时 S1 上的端口都为指定端口，且都处于转发状态，为根交换机。S3 上的 E 0/0/2 为替代端口，处于丢弃状态。MSTID，即 MSTP 的实例 ID，三台交换机上目前都为 0，即在默认情况下，所有 VLAN 都处于 MSTP 实例 0 中。

假如网络管理员配置 STP 模式为 RSTP，最终选举出来的根交换机及被阻塞的端口等结果将和目前 MSTP 的选举结果一致，即在 MSTP 的单个实例中，选举规则与 RSTP 一致，端口角色和状态与 RSTP 也一致。

在 HR 部门的 PC-2 上持续发送 ping 包至 PC-1，在 IT 部门的 PC-4 上持续发送 ping 包至 PC-3。

```
PC>ping 192.168.10.1 -t
Ping 192.168.10.1: 32 data bytes, Press Ctrl_C to break
From 192.168.10.1: bytes=32 seq=1 ttl=128 time=31 ms
From 192.168.10.1: bytes=32 seq=2 ttl=128 time=47 ms
……

PC>ping 192.168.20.1 -t
Ping 192.168.20.1: 32 data bytes, Press Ctrl_C to break
From 192.168.20.1: bytes=32 seq=1 ttl=128 time=47 ms
From 192.168.20.1: bytes=32 seq=2 ttl=128 time=62 ms
……
```

同时，在 S3 的 E 0/0/1 接口上抓包观察，如图 4-6 所示。

No.	Time	Source	Destination	Protocol	Length	Info
31	6.755000	192.168.10.1	192.168.10.2	ICMP	78	Echo (ping) reply id=0x74a3, seq=20/5120, ttl=128
32	7.301000	192.168.20.2	192.168.20.1	ICMP	78	Echo (ping) request id=0x75a3, seq=18/4608, ttl=128
33	7.332000	192.168.20.1	192.168.20.2	ICMP	78	Echo (ping) reply id=0x75a3, seq=18/4608, ttl=128
34	7.784000	192.168.10.2	192.168.10.1	ICMP	78	Echo (ping) request id=0x75a3, seq=21/5376, ttl=128
35	7.784000	192.168.10.1	192.168.10.2	ICMP	78	Echo (ping) reply id=0x75a3, seq=21/5376, ttl=128
36	8.174000	HuaweiTe_4d:0f:cf	Spanning-tree-(for-bridges	STP	119	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 0 Port = 0x8002
37	8.346000	192.168.20.2	192.168.20.1	ICMP	78	Echo (ping) request id=0x76a3, seq=19/4864, ttl=128
38	8.362000	192.168.20.1	192.168.20.2	ICMP	78	Echo (ping) reply id=0x76a3, seq=19/4864, ttl=128
39	8.814000	192.168.10.2	192.168.10.1	ICMP	78	Echo (ping) request id=0x76a3, seq=22/5632, ttl=128
40	8.814000	192.168.10.1	192.168.10.2	ICMP	78	Echo (ping) reply id=0x76a3, seq=22/5632, ttl=128
41	9.391000	192.168.20.2	192.168.20.1	ICMP	78	Echo (ping) request id=0x77a3, seq=20/5120, ttl=128
42	9.438000	192.168.20.1	192.168.20.2	ICMP	78	Echo (ping) reply id=0x77a3, seq=20/5120, ttl=128
43	9.844000	192.168.10.2	192.168.10.1	ICMP	78	Echo (ping) request id=0x77a3, seq=23/5888, ttl=128
44	9.859000	192.168.10.1	192.168.10.2	ICMP	78	Echo (ping) reply id=0x77a3, seq=23/5888, ttl=128
45	10.390000	HuaweiTe_4d:0f:cf	Spanning-tree-(for-bridges	STP	119	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 0 Port = 0x8002
46	10.436000	192.168.20.2	192.168.20.1	ICMP	78	Echo (ping) request id=0x78a3, seq=21/5376, ttl=128
47	10.452000	192.168.20.1	192.168.20.2	ICMP	78	Echo (ping) reply id=0x78a3, seq=21/5376, ttl=128
48	10.873000	192.168.10.2	192.168.10.1	ICMP	78	Echo (ping) request id=0x78a3, seq=24/6144, ttl=128

```
⊞ Frame 43: 78 bytes on wire (624 bits), 78 bytes captured (624 bits)
⊞ Ethernet II, Src: HuaweiTe_cf:c5:4e (54:89:98:cf:c5:4e), Dst: HuaweiTe_cf:40:0e (54:89:98:cf:40:0e)
⊞ 802.1Q Virtual LAN, PRI: 0, CFI: 0, ID: 10
⊞ Internet Protocol Version 4, Src: 192.168.10.2 (192.168.10.2), Dst: 192.168.10.1 (192.168.10.1)
⊞ Internet Control Message Protocol
```

图 4-6 抓包观察

可以观察到，目前 VLAN 10 和 VLAN 20 的数据包都从 S2 的接口 E 0/0/1 转发。

在 S3 的 E 0/0/2 接口上抓包观察，如图 4-7 所示。

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	119	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 1 Port = 0x8001
2	2.231000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	119	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 1 Port = 0x8001
3	4.462000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	119	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 1 Port = 0x8001
4	6.692000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	119	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 1 Port = 0x8001
5	8.939000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	119	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 1 Port = 0x8001
6	11.170000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	119	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 1 Port = 0x8001
7	13.401000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	119	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 1 Port = 0x8001
8	15.631000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	119	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 1 Port = 0x8001
9	17.862000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	119	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 1 Port = 0x8001
10	20.093000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	119	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 1 Port = 0x8001
11	22.324000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	119	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 1 Port = 0x8001
12	24.555000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	119	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 1 Port = 0x8001
13	26.770000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	119	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 1 Port = 0x8001
14	29.001000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	119	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 1 Port = 0x8001
15	31.231000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	119	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 1 Port = 0x8001
16	33.431000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	119	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 1 Port = 0x8001
17	35.662000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	119	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 1 Port = 0x8001

```
⊞ Frame 1: 119 bytes on wire (952 bits), 119 bytes captured (952 bits)
⊞ IEEE 802.3 Ethernet
⊞ Logical-Link Control
⊞ Spanning Tree Protocol
```

图 4-7 抓包观察

可以观察到，在 S3 的 E 0/0/2 接口上，没有任何数据包转发，只接收到上行接口周期发送的 BPDU。

此时 S2 与 S3 间的链路完全处于闲置状态，造成了资源的浪费，也导致了 S1 与 S3 间链路上数据转发任务繁重，易引起拥塞丢包。为了能够有效地利用链路资源，可以通过配置 MSTP 的多实例来实现。

关闭 PC 上的 ping 测试。

3. 配置 MSTP 多实例

MSTP 网络由一个或者多个 MST 域组成，每个 MST 域中可以包含一个或多个 MSTI，即 MST 实例。MST 域中含有一张 VLAN 映射表，描述了 VLAN 与 MSTI 之间的映射关系，默认情况下所有 VLAN 都映射到 MSTI 0 中。MSTI 之间彼此独立。

在 S1 上配置 MSTP 的多实例。使用 **stp region-configuration** 命令进入 MST 域视图。

```
[S1]stp region-configuration
[S1-mst-region]
```

使用 **region-name** 命令配置 MST 域名为 huawei。

```
[S1-mst-region]region-name huawei
```

使用 **revision-level** 命令配置 MSTP 的修订级别为 1。

```
[S1-mst-region]revision-level 1
```

使用 **instance** 命令指定 VLAN 10 映射到 MSTI 1，指定 VLAN 20 映射到 MSTI 2。

```
[S1-mst-region]instance 1 vlan 10
[S1-mst-region]instance 2 vlan 20
```

使用 **active region-configuration** 命令激活 MST 域配置。

```
[S1-mst-region]active region-configuration
Info: This operation may take a few seconds. Please wait for a moment...done.
```

在 S2、S3 上做同样配置，但是注意，在同一 MST 域中，必须具有相同域名、修订级别以及 VLAN 到 MSTI 的映射关系。

```
[S2]stp region-configuration
[S2-mst-region]region-name huawei
[S2-mst-region]revision-level 1
[S2-mst-region]instance 1 vlan 10
[S2-mst-region]instance 2 vlan 20
[S2-mst-region]active region-configuration

[S3]stp region-configuration
[S3-mst-region]region-name huawei
[S3-mst-region]revision-level 1
[S3-mst-region]instance 1 vlan 10
[S3-mst-region]instance 2 vlan 20
[S3-mst-region]active region-configuration
```

配置完成后，在 S1、S2、S3 上使用 **display stp region-configuration** 命令查看交换机上当前生效的 MST 域配置信息。

```
[S1]display stp region-configuration
 Oper configuration
   Format selector      :0
   Region name          :huawei
   Revision level       :1
   Instance    VLANs Mapped
     0         1 to 9, 11 to 19, 21 to 4094
     1         10
     2         20

[S2]display stp region-configuration
 Oper configuration
   Format selector      :0
   Region name          :huawei
   Revision level       :1
   Instance    VLANs Mapped
     0         1 to 9, 11 to 19, 21 to 4094
     1         10
     2         20

[S3]display stp region-configuration
 Oper configuration
   Format selector      :0
   Region name          :huawei
   Revision level       :1
```

```
Instance    VLANs Mapped
   0        1 to 9, 11 to 19, 21 to 4094
   1        10
   2        20
```

可以观察到，所有交换机上的 MST 域名都为 huawei，修订版本号都为 1，且 VLAN 与实例间的映射关系相同，其中除 VLAN 10 与 20 之外，其余 VLAN 都属于实例 0 中。

MSTP 多实例配置完成后，在 HR 部门的 PC-2 上持续发送 ping 包至 PC-1（使用 ping 192.168.10.1-t 命令），在 IT 部门的 PC-4 上持续发送 ping 包至 PC-3（使用 ping 192.168.20.1 –t 命令）。同时，在 S3 的 E 0/0/1 接口上抓包观察，如图 4-8 所示。

```
 1 0.000000 192.168.20.2     192.168.20.1                ICMP   78 Echo (ping) request  id=0xf6c1, seq=10/2560, ttl=128
 2 0.047000 192.168.20.1     192.168.20.2                ICMP   78 Echo (ping) reply    id=0xf6c1, seq=10/2560, ttl=128
 3 0.873000 192.168.10.2     192.168.10.1                ICMP   78 Echo (ping) request  id=0xf6c1, seq=13/3328, ttl=128
 4 0.889000 192.168.10.1     192.168.10.2                ICMP   78 Echo (ping) reply    id=0xf6c1, seq=13/3328, ttl=128
 5 1.045000 192.168.20.2     192.168.20.1                ICMP   78 Echo (ping) request  id=0xf7c1, seq=11/2816, ttl=128
 6 1.092000 192.168.20.1     192.168.20.2                ICMP   78 Echo (ping) reply    id=0xf7c1, seq=11/2816, ttl=128
 7 1.248000 HuaweiTe_4d:0f:cfSpanning-tree-(for-bridges STP   151 MST. Root = 32768/0/4c:1f:cc:4d:0f:cf  Cost = 0  Port = 0x8002
 8 1.903000 192.168.10.2     192.168.10.1                ICMP   78 Echo (ping) request  id=0xf8c1, seq=14/3584, ttl=128
 9 1.903000 192.168.10.1     192.168.10.2                ICMP   78 Echo (ping) reply    id=0xf8c1, seq=14/3584, ttl=128
10 2.090000 192.168.20.2     192.168.20.1                ICMP   78 Echo (ping) request  id=0xf8c1, seq=12/3072, ttl=128
11 2.137000 192.168.20.1     192.168.20.2                ICMP   78 Echo (ping) reply    id=0xf8c1, seq=12/3072, ttl=128
12 2.933000 192.168.10.2     192.168.10.1                ICMP   78 Echo (ping) request  id=0xf9c1, seq=15/3840, ttl=128
13 2.933000 192.168.10.1     192.168.10.2                ICMP   78 Echo (ping) reply    id=0xf9c1, seq=15/3840, ttl=128
14 3.151000 192.168.20.2     192.168.20.1                ICMP   78 Echo (ping) request  id=0xf9c1, seq=13/3328, ttl=128
15 3.198000 192.168.20.1     192.168.20.2                ICMP   78 Echo (ping) reply    id=0xf9c1, seq=13/3328, ttl=128
16 3.369000 HuaweiTe_4d:0f:cfSpanning-tree-(for-bridges STP   151 MST. Root = 32768/0/4c:1f:cc:4d:0f:cf  Cost = 0  Port = 0x8002
17 3.947000 192.168.10.2     192.168.10.1                ICMP   78 Echo (ping) request  id=0xfac1, seq=16/4096, ttl=128

⊞ Frame 1: 78 bytes on wire (624 bits), 78 bytes captured (624 bits)
⊞ Ethernet II, Src: HuaweiTe_cf:d1:4d (54:89:98:cf:d1:4d), Dst: HuaweiTe_cf:d8:74 (54:89:98:cf:d8:74)
⊞ 802.1Q Virtual LAN, PRI: 0, CFI: 0, ID: 20
⊞ Internet Protocol Version 4, Src: 192.168.20.2 (192.168.20.2), Dst: 192.168.20.1 (192.168.20.1)
⊞ Internet Control Message Protocol
```

图 4-8　抓包现象

可以观察到，目前 VLAN 10 和 VLAN 20 的数据包仍然从 E 0/0/1 转发。

在 S3 的 E 0/0/2 接口上抓包观察，如图 4-9 所示。

```
No. Time      Source             Destination                 Protocol Length Info
 6 10.905000 HuaweiTe_50:2e:9c Spanning-tree-(for-bridges STP  151 MST. Root = 32768/0/4c:1f:cc:4d:0f:cf  Cost = 0  Port = 0x8001
 7 13.136000 HuaweiTe_50:2e:9c Spanning-tree-(for-bridges STP  151 MST. Root = 32768/0/4c:1f:cc:4d:0f:cf  Cost = 0  Port = 0x8001
 8 15.351000 HuaweiTe_50:2e:9c Spanning-tree-(for-bridges STP  151 MST. Root = 32768/0/4c:1f:cc:4d:0f:cf  Cost = 0  Port = 0x8001
 9 17.566000 HuaweiTe_50:2e:9c Spanning-tree-(for-bridges STP  151 MST. Root = 32768/0/4c:1f:cc:4d:0f:cf  Cost = 0  Port = 0x8001
10 19.781000 HuaweiTe_50:2e:9c Spanning-tree-(for-bridges STP  151 MST. Root = 32768/0/4c:1f:cc:4d:0f:cf  Cost = 0  Port = 0x8001
11 21.996000 HuaweiTe_50:2e:9c Spanning-tree-(for-bridges STP  151 MST. Root = 32768/0/4c:1f:cc:4d:0f:cf  Cost = 0  Port = 0x8001
12 24.212000 HuaweiTe_50:2e:9c Spanning-tree-(for-bridges STP  151 MST. Root = 32768/0/4c:1f:cc:4d:0f:cf  Cost = 0  Port = 0x8001
13 26.427000 HuaweiTe_50:2e:9c Spanning-tree-(for-bridges STP  151 MST. Root = 32768/0/4c:1f:cc:4d:0f:cf  Cost = 0  Port = 0x8001
14 28.642000 HuaweiTe_50:2e:9c Spanning-tree-(for-bridges STP  151 MST. Root = 32768/0/4c:1f:cc:4d:0f:cf  Cost = 0  Port = 0x8001
15 30.842000 HuaweiTe_50:2e:9c Spanning-tree-(for-bridges STP  151 MST. Root = 32768/0/4c:1f:cc:4d:0f:cf  Cost = 0  Port = 0x8001
16 33.057000 HuaweiTe_50:2e:9c Spanning-tree-(for-bridges STP  151 MST. Root = 32768/0/4c:1f:cc:4d:0f:cf  Cost = 0  Port = 0x8001
17 35.225000 HuaweiTe_50:2e:9c Spanning-tree-(for-bridges STP  151 MST. Root = 32768/0/4c:1f:cc:4d:0f:cf  Cost = 0  Port = 0x8001
18 37.456000 HuaweiTe_50:2e:9c Spanning-tree-(for-bridges STP  151 MST. Root = 32768/0/4c:1f:cc:4d:0f:cf  Cost = 0  Port = 0x8001

⊞ Frame 1: 151 bytes on wire (1208 bits), 151 bytes captured (1208 bits)
⊞ IEEE 802.3 Ethernet
⊞ Logical-Link Control
⊞ Spanning Tree Protocol
```

图 4-9　抓包现象

可以观察到，在 E 0/0/2 接口上，仍然没有任何数据包转发，只有接收到的上行接口周期发送的 BPDU。

关闭 PC 上的 ping 测试。

现在已经配置了 MSTP 多实例，但由于每个 MSTP 实例都进行独立的生成树计算，所以在默认不变动任何生成树参数的情况下，其实每棵生成树的选举结果是一致的。

在 S1、S2、S3 上使用 **display stp instance 0 brief** 命令查看默认实例 0 中的生成树

状态和统计的摘要信息。

```
<S1>display stp instance 0 brief
 MSTID  Port                Role  STP State     Protection
   0    Ethernet0/0/1       DESI  FORWARDING    NONE
   0    Ethernet0/0/2       DESI  FORWARDING    NONE
   0    Ethernet0/0/3       DESI  FORWARDING    NONE

<S2>display stp instance 0 brief
 MSTID  Port                Role  STP State     Protection
   0    Ethernet0/0/1       DESI  FORWARDING    NONE
   0    Ethernet0/0/2       ROOT  FORWARDING    NONE
   0    Ethernet0/0/3       DESI  FORWARDING    NONE

<S3>display stp instance 0 brief
 MSTID  Port                Role  STP State     Protection
   0    Ethernet0/0/1       ROOT  FORWARDING    NONE
   0    Ethernet0/0/2       ALTE  DISCARDING    NONE
   0    Ethernet0/0/3       DESI  FORWARDING    NONE
   0    Ethernet0/0/4       DESI  FORWARDING    NONE
```

在 S1、S2、S3 上使用 **display stp instance 1 brief** 命令查看实例 1 中的生成树状态和统计的摘要信息。

```
<S1>display stp instance 1 brief
 MSTID  Port                Role  STP State     Protection
   1    Ethernet0/0/1       DESI  FORWARDING    NONE
   1    Ethernet0/0/2       DESI  FORWARDING    NONE
   1    Ethernet0/0/3       DESI  FORWARDING    NONE

<S2>display stp instance 1 brief
 MSTID  Port                Role  STP State     Protection
   1    Ethernet0/0/1       DESI  FORWARDING    NONE
   1    Ethernet0/0/2       ROOT  FORWARDING    NONE

<S3>display stp instance 1 brief
 MSTID  Port                Role  STP State     Protection
   1    Ethernet0/0/1       ROOT  FORWARDING    NONE
   1    Ethernet0/0/2       ALTE  DISCARDING    NONE
   1    Ethernet0/0/3       DESI  FORWARDING    NONE
```

在 S1、S2、S3 上使用 **display stp instance 2 brief** 命令查看实例 2 中的生成树状态和统计的摘要信息。

```
<S1>display stp instance 2 brief
 MSTID  Port                Role  STP State     Protection
   2    Ethernet0/0/1       DESI  FORWARDING    NONE
   2    Ethernet0/0/2       DESI  FORWARDING    NONE

<S2>display stp instance 2 brief
 MSTID  Port                Role  STP State     Protection
   2    Ethernet0/0/1       DESI  FORWARDING    NONE
   2    Ethernet0/0/2       ROOT  FORWARDING    NONE
   2    Ethernet0/0/3       DESI  FORWARDING    NONE

<S3>display stp instance 2 brief
 MSTID  Port                Role  STP State     Protection
   2    Ethernet0/0/1       ROOT  FORWARDING    NONE
   2    Ethernet0/0/2       ALTE  DISCARDING    NONE
```

```
  2    Ethernet0/0/4              DESI   DISCARDING        NONE
```

可以观察到，在以上 3 个实例中，选举结果是一致的，都是 S3 的 E 0/0/2 接口处于 Discarding 状态。

现在要实现 S2 与 S3 间的链路被利用，可以在实例 1 中，保持目前生成树选举结果不变，即使得 VLAN 10 中的 HR 部门内的流量通过 S1 与 S3 间的链路转发。在实例 2 中，配置使得 S2 成为根交换机，阻塞 S1 与 S3 间的链路，即使得 VLAN 20 中的 IT 部门的流量通过 S2 与 S3 间的链路转发。

在 S2 上使用 **stp instance priority** 命令配置其成为实例 2 中的根交换机。

```
[S2]stp instance 2 priority 0
```

配置完成后，在 S1、S2、S3 上使用 **display stp instance 2 brief** 命令查看实例 2 中的生成树状态和统计的摘要信息。

```
<S1>display stp instance 2 brief
 MSTID  Port                      Role   STP State         Protection
   2    Ethernet0/0/1             ROOT   FORWARDING        NONE
   2    Ethernet0/0/2             DESI   FORWARDING        NONE

<S2>display stp instance 2 brief
 MSTID  Port                      Role   STP State         Protection
   2    Ethernet0/0/1             DESI   FORWARDING        NONE
   2    Ethernet0/0/2             DESI   FORWARDING        NONE
   2    Ethernet0/0/3             DESI   FORWARDING        NONE

<S3>display stp instance 2 brief
 MSTID  Port                      Role   STP State         Protection
   2    Ethernet0/0/1             ALTE   DISCARDING        NONE
   2    Ethernet0/0/2             ROOT   FORWARDING        NONE
   2    Ethernet0/0/4             DESI   FORWARDING        NONE
```

可以观察到，此时 S2 成为了实例 2 中的根交换机，所有端口都为指定端口，而 S3 的 E 0/0/1 接口为替代端口，即 S1 与 S3 间的链路现已阻塞。

在 HR 部门的 PC-2 上持续发送 ping 包至 PC-1（使用 ping 192.168.10.1 –t 命令），在 IT 部门的 PC-4 上持续发送 ping 包至 PC-3（使用 ping 192.168.20.1 –t 命令）。同时，在 S3 的 E 0/0/1 接口上抓包观察，如图 4-10 所示。

No.	Time	Source	Destination	Protocol	Length	Info
28	10.670000	HuaweiTe_4d:0f:cf	Spanning-tree-(for-bridges	STP	151	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 0 Port = 0x8002
29	11.435000	192.168.10.2	192.168.10.1	ICMP	78	Echo (ping) request id=0xcacc, seq=22/5632, ttl=128
30	11.450000	192.168.10.1	192.168.10.2	ICMP	78	Echo (ping) reply id=0xcacc, seq=22/5632, ttl=128
31	12.480000	192.168.10.2	192.168.10.1	ICMP	78	Echo (ping) request id=0xcbcc, seq=23/5888, ttl=128
32	12.495000	192.168.10.1	192.168.10.2	ICMP	78	Echo (ping) reply id=0xcbcc, seq=23/5888, ttl=128
33	12.792000	HuaweiTe_4d:0f:cf	Spanning-tree-(for-bridges	STP	151	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 0 Port = 0x8002
34	13.509000	192.168.10.2	192.168.10.1	ICMP	78	Echo (ping) request id=0xcccc, seq=24/6144, ttl=128
35	13.509000	192.168.10.1	192.168.10.2	ICMP	78	Echo (ping) reply id=0xcccc, seq=24/6144, ttl=128
36	14.555000	192.168.10.2	192.168.10.1	ICMP	78	Echo (ping) request id=0xcdcc, seq=25/6400, ttl=128
37	14.555000	192.168.10.1	192.168.10.2	ICMP	78	Echo (ping) reply id=0xcdcc, seq=25/6400, ttl=128
38	14.929000	HuaweiTe_4d:0f:cf	Spanning-tree-(for-bridges	STP	151	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 0 Port = 0x8002
39	15.569000	192.168.10.2	192.168.10.1	ICMP	78	Echo (ping) request id=0xcecc, seq=26/6656, ttl=128
40	15.584000	192.168.10.1	192.168.10.2	ICMP	78	Echo (ping) reply id=0xcecc, seq=26/6656, ttl=128
41	16.598000	192.168.10.2	192.168.10.1	ICMP	78	Echo (ping) request id=0xcfcc, seq=27/6912, ttl=128
42	16.629000	192.168.10.1	192.168.10.2	ICMP	78	Echo (ping) reply id=0xcfcc, seq=27/6912, ttl=128
43	17.051000	HuaweiTe_4d:0f:cf	Spanning-tree-(for-bridges	STP	151	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 0 Port = 0x8002

```
⊞ Frame 1: 151 bytes on wire (1208 bits), 151 bytes captured (1208 bits)
⊞ IEEE 802.3 Ethernet
⊞ Logical-Link Control
⊞ Spanning Tree Protocol
```

图 4-10 抓包观察

可以观察到，目前 VLAN 10 的流量都从 S3 的 E 0/0/1 接口转发。

在 S3 的 E 0/0/2 接口上抓包观察，如图 4-11 所示。

No.	Time	Source	Destination	Protocol	Length	Info
2	0.016000	192.168.20.1	192.168.20.2	ICMP	78	Echo (ping) reply id=0xc6cc, seq=16/4096, ttl=128
3	1.014000	192.168.20.2	192.168.20.1	ICMP	78	Echo (ping) request id=0xc7cc, seq=17/4352, ttl=128
4	1.014000	192.168.20.1	192.168.20.2	ICMP	78	Echo (ping) reply id=0xc7cc, seq=17/4352, ttl=128
5	1.373000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	151	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 0 Port = 0x8001
6	2.059000	192.168.20.2	192.168.20.1	ICMP	78	Echo (ping) request id=0xc8cc, seq=18/4608, ttl=128
7	2.075000	192.168.20.1	192.168.20.2	ICMP	78	Echo (ping) reply id=0xc8cc, seq=18/4608, ttl=128
8	3.089000	192.168.20.2	192.168.20.1	ICMP	78	Echo (ping) request id=0xc9cc, seq=19/4864, ttl=128
9	3.104000	192.168.20.1	192.168.20.2	ICMP	78	Echo (ping) reply id=0xc9cc, seq=19/4864, ttl=128
10	3.510000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	151	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 0 Port = 0x8001
11	4.118000	192.168.20.2	192.168.20.1	ICMP	78	Echo (ping) request id=0xcacc, seq=20/5120, ttl=128
12	4.150000	192.168.20.1	192.168.20.2	ICMP	78	Echo (ping) reply id=0xcacc, seq=20/5120, ttl=128
13	5.179000	192.168.20.2	192.168.20.1	ICMP	78	Echo (ping) request id=0xcbcc, seq=21/5376, ttl=128
14	5.179000	192.168.20.1	192.168.20.2	ICMP	78	Echo (ping) reply id=0xcbcc, seq=21/5376, ttl=128
15	5.632000	HuaweiTe_50:2e:9c	Spanning-tree-(for-bridges	STP	151	MST. Root = 32768/0/4c:1f:cc:4d:0f:cf Cost = 0 Port = 0x8001
16	6.193000	192.168.20.2	192.168.20.1	ICMP	78	Echo (ping) request id=0xcccc, seq=22/5632, ttl=128
17	6.209000	192.168.20.1	192.168.20.2	ICMP	78	Echo (ping) reply id=0xcccc, seq=22/5632, ttl=128
18	7.207000	192.168.20.2	192.168.20.1	ICMP	78	Echo (ping) request id=0xcdcc, seq=23/5888, ttl=128

```
⊞ Frame 1: 78 bytes on wire (624 bits), 78 bytes captured (624 bits)
⊞ Ethernet II, Src: HuaweiTe_cf:d1:4d (54:89:98:cf:d1:4d), Dst: HuaweiTe_cf:d8:74 (54:89:98:cf:d8:74)
⊞ 802.1Q Virtual LAN, PRI: 0, CFI: 0, ID: 20
⊞ Internet Protocol Version 4, Src: 192.168.20.2 (192.168.20.2), Dst: 192.168.20.1 (192.168.20.1)
⊞ Internet Control Message Protocol
```

图 4-11　抓包观察

可以观察到，目前 VLAN 20 的流量都从 E 0/0/2 接口转发。

至此，完成了 MSTP 的多实例的配置，并达到了流量分担的目的，有效地利用了网络资源，也同时使得 S3 的两条上行链路可以互相备份。

思考

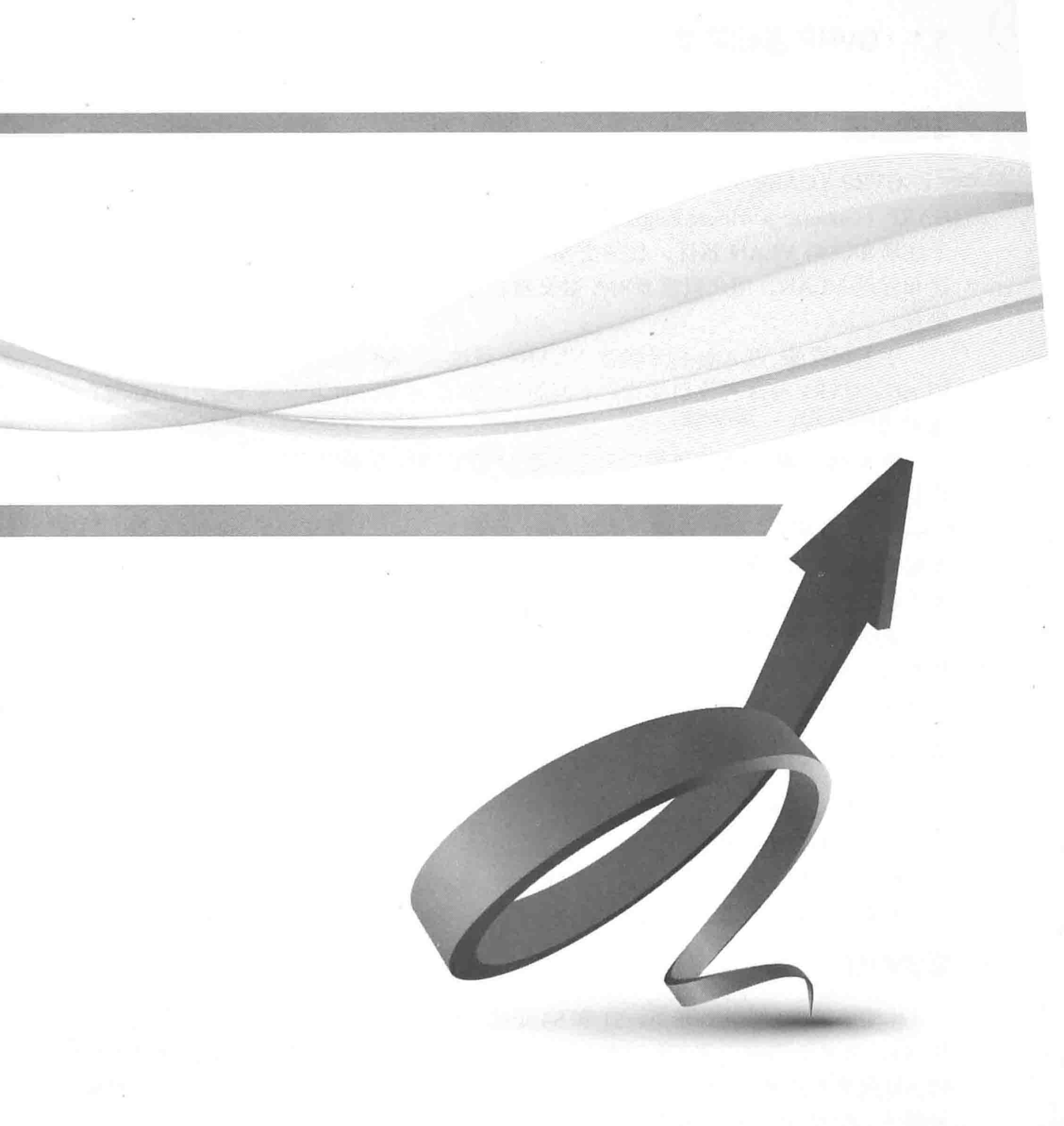
当 MSTP 和 RSTP 混合使用的时候，如何选举根桥？

第5章 其他交换技术

5.1 GVRP基础配置

5.2 Smart Link与Monitor Link

5.3 配置Eth-Trunk链路聚合



5.1 GVRP 基础配置

原理概述

GVRP（GARP VLAN Registration Protocol），中文名为 GARP VLAN 注册协议，是 GARP（Generic Attribute Registration Protocol，通用属性注册协议）的一种应用，用于注册和注销 VLAN 属性。使得交换机之间能够相互交换 VLAN 配置信息，动态创建和管理 VLAN。用户只需要对少数交换机进行 VLAN 配置即可动态地传播 VLAN 信息。

手工配置的 VLAN 称为静态 VLAN，通过 GVRP 协议创建的 VLAN 称为动态 VLAN。GVRP 有 3 种注册模式，不同的模式对静态 VLAN 和动态 VLAN 的处理方式也不同。

- Normal 模式：允许该接口动态注册、注销 VLAN，传播动态 VLAN 以及静态 VLAN 信息；
- Fixed 模式：禁止该接口动态注册、注销 VLAN，只传播静态 VLAN 信息。即被设置成为该模式下的 Trunk 接口，即使允许所有 VLAN 通过，实际通过的 VLAN 也只能是手动配置的那部分；
- Forbidden 模式：禁止该接口动态注册、注销 VLAN，不传播任何除 VLAN 1 以外的任何 VLAN 信息。即被设置成为该模式下的 Trunk 接口，即使允许所有 VLAN 通过，实际通过的 VLAN 也只能是 VLAN 1。

实验目的

- 理解 GVRP 的应用场景
- 掌握 GVRP 的配置
- 理解 GVRP 不同注册模式的区别
- 掌握 GVRP 配置不同注册模式的方法

实验内容

本实验模拟企业网络场景。S1 和 S4 是接入层交换机，分别连接到汇聚层交换机 S2 和 S3，公司不同部门员工通过接入层交换机连接到网络。现在需要在交换机上划分 VLAN 隔离不同部门，但考虑到部门较多，且随着发展，网络情况可能会越来越复杂，采用手工配置 VLAN 的方式工作量会非常大，而且容易导致配置错误。此时可以通过 GVRP 的 VLAN 自动注册功能完成 VLAN 的配置。

实验拓扑

GVRP 基础配置的拓扑如图 5-1 所示。

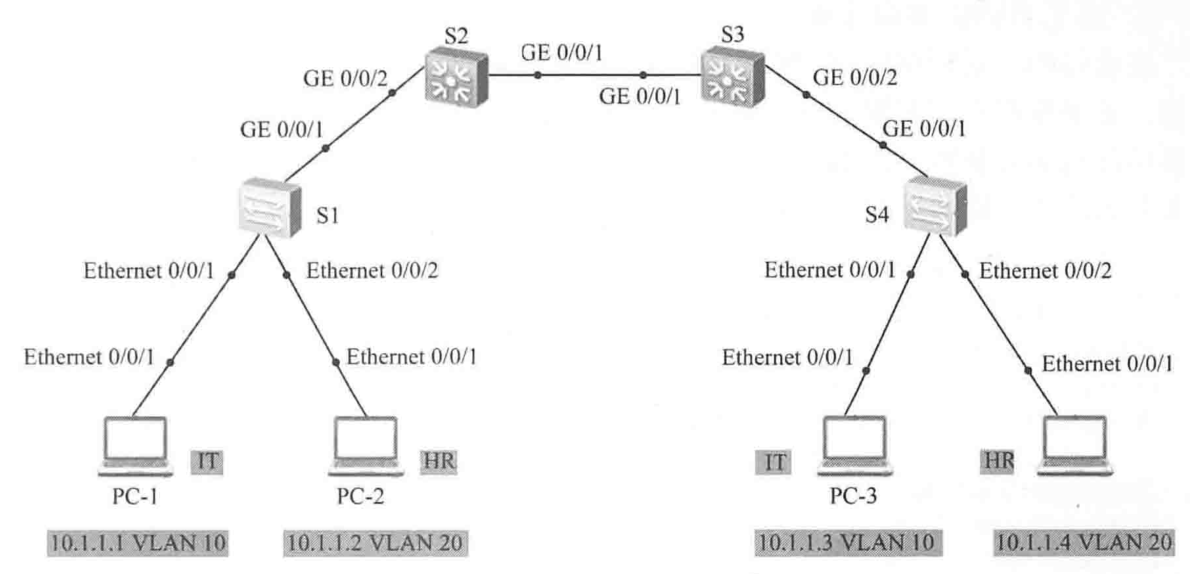


图 5-1　GVRP 基础配置拓扑

实验编址

实验编址见表 5-1。

表 5-1　　实验编址

设备	接口	IP 地址	子网掩码	默认网关
PC-1	Ethernet 0/0/1	10.1.1.1	255.255.255.0	N/A
PC-2	Ethernet 0/0/1	10.1.1.2	255.255.255.0	N/A
PC-3	Ethernet 0/0/1	10.1.1.3	255.255.255.0	N/A
PC-4	Ethernet 0/0/1	10.1.1.4	255.255.255.0	N/A

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 地址配置，并使用 **ping** 命令检测各直连链路的连通性，在没有完成划分 VLAN 之前各 PC 都属于默认的 VLAN 1，之间都能互通。测试 PC-1 与 PC-2 间的连通性。

```
PC>ping 10.1.1.2
Ping 10.1.1.2: 32 data bytes, Press Ctrl_C to break
From 10.1.1.2: bytes=32 seq=1 ttl=128 time=16 ms
From 10.1.1.2: bytes=32 seq=2 ttl=128 time=31 ms
From 10.1.1.2: bytes=32 seq=3 ttl=128 time<1 ms
From 10.1.1.2: bytes=32 seq=4 ttl=128 time=15 ms
From 10.1.1.2: bytes=32 seq=5 ttl=128 time=15 ms
--- 10.1.1.2 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
0.00% packet loss
  round-trip min/avg/max = 0/15/31 ms
```

其余主机间的连通性测试省略。

2. 配置 GVRP 单向注册

在公司的二层网络中，有 IT 和 HR 两个不同的部门，需要将它们划分到不同的 VLAN 中去。如果按照常规配置方法，要手工在所有交换机上都创建相应的 VLAN。后续如果有新的部门需要新增 VLAN，或者二层网络整改，都要随之修改 VLAN 配置，配置量非常大且易出错，现网络管理员采用 GVRP 来完成 VLAN 配置。

将 4 台交换机之间所互连的接口（连接 PC 的接口除外）的接口类型都配置为 Trunk，并允许所有 VLAN 通过。

```
[S1]interface GigabitEthernet 0/0/1
[S1-GigabitEthernet0/0/1]port link-type trunk
[S1-GigabitEthernet0/0/1]port trunk allow-pass vlan all

[S2]interface GigabitEthernet 0/0/1
[S2-GigabitEthernet0/0/1]port link-type trunk
[S2-GigabitEthernet0/0/1]port trunk allow-pass vlan all
[S2-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2
[S2-GigabitEthernet0/0/2]port link-type trunk
[S2-GigabitEthernet0/0/2]port trunk allow-pass vlan all

[S3]interface GigabitEthernet 0/0/1
[S3-GigabitEthernet0/0/1]port link-type trunk
[S3-GigabitEthernet0/0/1]port trunk allow-pass vlan all
[S3-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2
[S3-GigabitEthernet0/0/2]port link-type trunk
[S3-GigabitEthernet0/0/2]port trunk allow-pass vlan all

[S4]interface GigabitEthernet 0/0/1
[S4-GigabitEthernet0/0/1]port link-type trunk
[S4-GigabitEthernet0/0/1]port trunk allow-pass vlan all
```

在 S1 上创建 VLAN 10 和 VLAN 20，并把连接 PC 的接口类型配置为 Access，划入到相应的 VLAN 中。

```
[S1]vlan 10
[S1-Vlan10]vlan 20
[S1-Vlan20]interface Ethernet0/0/1
[S1-Ethernet0/0/1]port link-type access
[S1-Ethernet0/0/1]port default vlan 10
[S1-Ethernet0/0/1]interface Ethernet0/0/2
[S1-Ethernet0/0/2]port link-type access
[S1-Ethernet0/0/2]port default vlan 20
```

在所有交换机上都开启 GVRP 功能，并在所有交换机两两互连的接口下也开启 GVRP 功能。GVRP 注册模式默认为 Normal 模式。

```
[S1]gvrp
[S1]interface GigabitEthernet 0/0/1
[S1-GigabitEthernet0/0/1]gvrp

[S2]gvrp
[S2]interface GigabitEthernet 0/0/1
[S2-GigabitEthernet0/0/1]gvrp
[S2-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2
[S2-GigabitEthernet0/0/2]gvrp
```

```
[S3]gvrp
[S3]interface GigabitEthernet 0/0/1
[S3-GigabitEthernet0/0/1]gvrp
[S3-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2
[S3-GigabitEthernet0/0/2]gvrp

[S4]gvrp
[S4]interface GigabitEthernet 0/0/1
[S4-GigabitEthernet0/0/1]gvrp
```

配置完成后，在 S2、S3、S4 上使用 **display vlan** 命令查看所有 VLAN 的相关信息。

```
[S2]display vlan
The total number of vlans is : 3
--------------------------------------------------------------------------------
U: Up;          D: Down;         TG: Tagged;          UT: Untagged;
MP: Vlan-mapping;                ST: Vlan-stacking;
#: ProtocolTransparent-vlan;    *: Management-vlan;
--------------------------------------------------------------------------------
VID  Type     Ports
--------------------------------------------------------------------------------
1    common  UT:GE0/0/1(U)      GE0/0/2(U)      GE0/0/3(D)      GE0/0/4(D)
                GE0/0/5(D)      GE0/0/6(D)      GE0/0/7(D)      GE0/0/8(D)
                GE0/0/9(D)      GE0/0/10(D)     GE0/0/11(D)     GE0/0/12(D)
                GE0/0/13(D)     GE0/0/14(D)     GE0/0/15(D)     GE0/0/16(D)
                GE0/0/17(D)     GE0/0/18(D)     GE0/0/19(D)     GE0/0/20(D)
                GE0/0/21(D)     GE0/0/22(D)     GE0/0/23(D)     GE0/0/24(D)
10   dynamic TG:GE0/0/2(U)
20   dynamic TG:GE0/0/2(U)
VID  Status  Property      MAC-LRN     Statistics  Description
1    enable  default       enable      disable     VLAN 0001
10   enable  default       enable      disable     VLAN 0010
20   enable  default       enable      disable     VLAN 0020

[S3]display vlan
The total number of vlans is : 3
--------------------------------------------------------------------------------
U: Up;          D: Down;         TG: Tagged;          UT: Untagged;
MP: Vlan-mapping;                ST: Vlan-stacking;
#: ProtocolTransparent-vlan;    *: Management-vlan;
--------------------------------------------------------------------------------
VID  Type     Ports
--------------------------------------------------------------------------------
1    common  UT:GE0/0/1(U)      GE0/0/2(U)      GE0/0/3(D)      GE0/0/4(D)
                GE0/0/5(D)      GE0/0/6(D)      GE0/0/7(D)      GE0/0/8(D)
                GE0/0/9(D)      GE0/0/10(D)     GE0/0/11(D)     GE0/0/12(D)
                GE0/0/13(D)     GE0/0/14(D)     GE0/0/15(D)     GE0/0/16(D)
                GE0/0/17(D)     GE0/0/18(D)     GE0/0/19(D)     GE0/0/20(D)
                GE0/0/21(D)     GE0/0/22(D)     GE0/0/23(D)     GE0/0/24(D)
10   dynamic TG:GE0/0/1(U)
20   dynamic TG:GE0/0/1(U)
VID  Status  Property      MAC-LRN     Statistics  Description
1    enable  default       enable      disable     VLAN 0001
10   enable  default       enable      disable     VLAN 0010
20   enable  default       enable      disable     VLAN 0020

[S4]display vlan
```

```
The total number of vlans is : 3
--------------------------------------------------------------------------------
U: Up;            D: Down;          TG: Tagged;          UT: Untagged;
MP: Vlan-mapping;                   ST: Vlan-stacking;
#: ProtocolTransparent-vlan;     *: Management-vlan;
--------------------------------------------------------------------------------
VID  Type     Ports
--------------------------------------------------------------------------------
1    common   UT:Eth0/0/1(U)    Eth0/0/2(U)     Eth0/0/3(D)     Eth0/0/4(D)
                 Eth0/0/5(D)    Eth0/0/6(D)     Eth0/0/7(D)     Eth0/0/8(D)
                 Eth0/0/9(D)    Eth0/0/10(D)    Eth0/0/11(D)    Eth0/0/12(D)
                 Eth0/0/13(D)   Eth0/0/14(D)    Eth0/0/15(D)    Eth0/0/16(D)
                 Eth0/0/17(D)   Eth0/0/18(D)    Eth0/0/19(D)    Eth0/0/20(D)
                 Eth0/0/21(D)   Eth0/0/22(D)    GE0/0/1(U)      GE0/0/2(D)
10   dynamic TG:GE0/0/1(U)
20   dynamic TG:GE0/0/1(U)
VID  Status  Property      MAC-LRN Statistics Description
1    enable  default       enable  disable    VLAN 0001
10   enable  default       enable  disable    VLAN 0010
20   enable  default       enable  disable    VLAN 0020
```

可以观察到，S2、S3、S4 都动态获得了 VLAN 10 和 VLAN 20。但是在 S2 上只有 GE 0/0/2 加入了这两个 VLAN，同样在 S3 上只有 GE 0/0/1 加入这两个 VLAN，在 S4 上只有 GE 0/0/1 加入了这两个 VLAN。这是因为此时在 S2、S3、S4 上只有左侧的端口收到 GVRP 的注册消息，此时只完成了单向注册。

由于 PC-1 与 PC-3 同属于 IT 部门，即 VLAN 10 内，现验证它们之间的连通性。

```
PC>ping 10.1.1.3
Ping 10.1.1.3: 32 data bytes, Press Ctrl_C to break
From 10.1.1.1: Destination host unreachable
From 10.1.1.1: Destination host unreachable
From 10.1.1.1: Destination host unreachable
From 10.1.1.1: Destination host unreachable
From 10.1.1.1: Destination host unreachable
……
```

发现无法通信，再次验证了此时只完成了单向注册。

3. 配置 GVRP 双向注册

现需要在 S4 上也手工创建 VLAN 10 和 VLAN 20，把连接 PC 的接口的模式配置为 Access，划入相应 VLAN 中。

```
<S4>system-view
[S4]vlan 10
[S4-Vlan10]vlan 20
[S4-Vlan20]interface Ethernet0/0/1
[S4-Ethernet0/0/1]port link-type access
[S4-Ethernet0/0/1]port default vlan 10
[S4-Ethernet0/0/1]interface Ethernet0/0/2
[S4-Ethernet0/0/2]port link-type access
[S4-Ethernet0/0/2]port default vlan 20
```

配置完成后，在 S2、S3 上再次使用 **display vlan** 命令查看。

```
[S2]display vlan
The total number of vlans is : 3
--------------------------------------------------------------------------------
U: Up;            D: Down;          TG: Tagged;          UT: Untagged;
```

```
MP: Vlan-mapping;                    ST: Vlan-stacking;
#: ProtocolTransparent-vlan;         *: Management-vlan;
--------------------------------------------------------------------------------
VID  Type     Ports
--------------------------------------------------------------------------------
1    common  UT:GE0/0/1(U)    GE0/0/2(U)     GE0/0/3(D)     GE0/0/4(D)
                GE0/0/5(D)     GE0/0/6(D)     GE0/0/7(D)     GE0/0/8(D)
                GE0/0/9(D)     GE0/0/10(D)    GE0/0/11(D)    GE0/0/12(D)
                GE0/0/13(D)    GE0/0/14(D)    GE0/0/15(D)    GE0/0/16(D)
                GE0/0/17(D)    GE0/0/18(D)    GE0/0/19(D)    GE0/0/20(D)
                GE0/0/21(D)    GE0/0/22(D)    GE0/0/23(D)    GE0/0/24(D)
10   dynamic TG:GE0/0/1(U)    GE0/0/2(U)
20   dynamic TG:GE0/0/1(U)    GE0/0/2(U)
VID  Status  Property     MAC-LRN     Statistics   Description
1    enable  default      enable      disable      VLAN 0001
10   enable  default      enable      disable      VLAN 0010
20   enable  default      enable      disable      VLAN 0020

[S3]display vlan
The total number of vlans is : 3
--------------------------------------------------------------------------------
U: Up;           D: Down;          TG: Tagged;          UT: Untagged;
MP: Vlan-mapping;                    ST: Vlan-stacking;
#: ProtocolTransparent-vlan;         *: Management-vlan;
--------------------------------------------------------------------------------
VID  Type     Ports
--------------------------------------------------------------------------------
1    common  UT:GE0/0/1(U)    GE0/0/2(U)     GE0/0/3(D)     GE0/0/4(D)
                GE0/0/5(D)     GE0/0/6(D)     GE0/0/7(D)     GE0/0/8(D)
                GE0/0/9(D)     GE0/0/10(D)    GE0/0/11(D)    GE0/0/12(D)
                GE0/0/13(D)    GE0/0/14(D)    GE0/0/15(D)    GE0/0/16(D)
                GE0/0/17(D)    GE0/0/18(D)    GE0/0/19(D)    GE0/0/20(D)
                GE0/0/21(D)    GE0/0/22(D)    GE0/0/23(D)    GE0/0/24(D)
10   dynamic TG:GE0/0/1(U)    GE0/0/2(U)
20   dynamic TG:GE0/0/1(U)    GE0/0/2(U)
VID  Status  Property     MAC-LRN     Statistics   Description
1    enable  default      enable      disable      VLAN 0001
10   enable  default      enable      disable      VLAN 0010
20   enable  default      enable      disable      VLAN 0020
```

可以观察到此时两台汇聚交换机的右侧接口也加入了 VLAN 10 和 VLAN 20。因为从右侧收到了 S4 的 GVRP 注册消息，此时完成了双向注册。

在 PC-1 上验证与 PC-3 之间的连通性。

```
PC>ping 10.1.1.3
Ping 10.1.1.3: 32 data bytes, Press Ctrl_C to break
From 10.1.1.3: bytes=32 seq=1 ttl=128 time=78 ms
From 10.1.1.3: bytes=32 seq=2 ttl=128 time=109 ms
From 10.1.1.3: bytes=32 seq=3 ttl=128 time=63 ms
From 10.1.1.3: bytes=32 seq=4 ttl=128 time=62 ms
From 10.1.1.3: bytes=32 seq=5 ttl=128 time=31 ms
--- 10.1.1.3 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
0.00% packet loss
  round-trip min/avg/max = 31/68/109 ms
```

此时可以正常通信，PC-2 与 PC-4 的连通性测试这里省略。

如果现在公司网络整改，需要添加新的汇聚交换机，或者替换新款设备，或者增删VLAN配置，都可以通过GVRP动态实现自动配置，不再需要手工配置。

4. 配置GVRP的Fixed模式

现在S3的GE 0/0/1接口下将GVRP的注册模式修改为Fixed模式。

```
[S3]interface GigabitEthernet 0/0/1
[S3-GigabitEthernet0/0/1]gvrp registration fixed
```

在S3上使用display vlan命令查看。

```
[S3]display vlan
The total number of vlans is : 3
--------------------------------------------------------------------------------
U: Up;          D: Down;          TG: Tagged;          UT: Untagged;
MP: Vlan-mapping;                 ST: Vlan-stacking;
#: ProtocolTransparent-vlan;      *: Management-vlan;
--------------------------------------------------------------------------------
VID  Type     Ports
--------------------------------------------------------------------------------
1    common   UT:GE0/0/1(U)     GE0/0/2(U)      GE0/0/3(D)      GE0/0/4(D)
                 GE0/0/5(D)     GE0/0/6(D)      GE0/0/7(D)      GE0/0/8(D)
                 GE0/0/9(D)     GE0/0/10(D)     GE0/0/11(D)     GE0/0/12(D)
                 GE0/0/13(D)    GE0/0/14(D)     GE0/0/15(D)     GE0/0/16(D)
                 GE0/0/17(D)    GE0/0/18(D)     GE0/0/19(D)     GE0/0/20(D)
                 GE0/0/21(D)    GE0/0/22(D)     GE0/0/23(D)     GE0/0/24(D)
10   dynamic  TG:GE0/0/2(U)
20   dynamic  TG:GE0/0/2(U)
VID  Status  Property     MAC-LRN   Statistics  Description
--------------------------------------------------------------------------------
1    enable  default      enable    disable     VLAN 0001
10   enable  default      enable    disable     VLAN 0010
20   enable  default      enable    disable     VLAN 0020
```

发现在GE 0/0/1接口已经无法动态学习到VLAN信息，这是由于Fixed模式下不允许动态VLAN在接口上注册，相应同部门内跨交换机的两台PC就无法通信。

这时的解决办法有两种，一种是在S3上手工创建VLAN 10和VLAN 20，另一种是恢复GE 0/0/1接口下GVRP注册模式为Normal模式，做相应配置即可。

5. 配置GVRP的Forbidden模式

在S2的GE 0/0/1接口下将GVRP的注册模式修改为Forbidden模式。

```
[S2]interface GigabitEthernet 0/0/1
[S2-GigabitEthernet0/0/1]gvrp registration forbidden
```

在S2上使用display vlan命令查看。

```
[S2]display vlan
The total number of vlans is : 3
--------------------------------------------------------------------------------
U: Up;          D: Down;          TG: Tagged;          UT: Untagged;
MP: Vlan-mapping;                 ST: Vlan-stacking;
#: ProtocolTransparent-vlan;      *: Management-vlan;
--------------------------------------------------------------------------------
VID  Type     Ports
--------------------------------------------------------------------------------
1    common   UT:GE0/0/1(U)     GE0/0/2(U)      GE0/0/3(D)      GE0/0/4(D)
                 GE0/0/5(D)     GE0/0/6(D)      GE0/0/7(D)      GE0/0/8(D)
```

```
                GE0/0/9(D)      GE0/0/10(D)     GE0/0/11(D)     GE0/0/12(D)
                GE0/0/13(D)     GE0/0/14(D)     GE0/0/15(D)     GE0/0/16(D)
                GE0/0/17(D)     GE0/0/18(D)     GE0/0/19(D)     GE0/0/20(D)
                GE0/0/21(D)     GE0/0/22(D)     GE0/0/23(D)     GE0/0/24(D)
10   common  TG:GE0/0/2(U)
20   common  TG:GE0/0/2(U)
VID  Status  Property      MAC-LRN    Statistics  Description
1    enable  default       enable     disable     VLAN 0001
10   enable  default       enable     disable     VLAN 0010
20   enable  default       enable     disable     VLAN 0020
```

可以观察到此时 VLAN 10、VLAN 20 中都没有 GE 0/0/1 接口加入。

在 S2 上手工创建 VLAN 10 和 VLAN 20，并使用 **display vlan** 命令查看。

```
[S2]vlan batch 10 20

[S2]display vlan
The total number of vlans is : 3
--------------------------------------------------------------------------------
U: Up;          D: Down;          TG: Tagged;          UT: Untagged;
MP: Vlan-mapping;                 ST: Vlan-stacking;
#: ProtocolTransparent-vlan;    *: Management-vlan;
--------------------------------------------------------------------------------
VID  Type    Ports
--------------------------------------------------------------------------------
1    common  UT:GE0/0/1(U)      GE0/0/2(U)      GE0/0/3(D)      GE0/0/4(D)
                GE0/0/5(D)      GE0/0/6(D)      GE0/0/7(D)      GE0/0/8(D)
                GE0/0/9(D)      GE0/0/10(D)     GE0/0/11(D)     GE0/0/12(D)
                GE0/0/13(D)     GE0/0/14(D)     GE0/0/15(D)     GE0/0/16(D)
                GE0/0/17(D)     GE0/0/18(D)     GE0/0/19(D)     GE0/0/20(D)
                GE0/0/21(D)     GE0/0/22(D)     GE0/0/23(D)     GE0/0/24(D)
10   common  TG:GE0/0/2(U)
20   common  TG:GE0/0/2(U)
VID  Status  Property      MAC-LRN    Statistics  Description
1    enable  default       enable     disable     VLAN 0001
10   enable  default       enable     disable     VLAN 0010
20   enable  default       enable     disable     VLAN 0020
```

结果还是一样，接口 GE 0/0/1 仍然没有加入到 VLAN 10 或 VLAN 20 中。这是因为 GE 0/0/1 接口下注册模式配置成了 Forbidden 模式后，将只允许 VLAN 1 通过。

思考

GVRP 能够应用在 Hybrid 类型的接口上吗？

5.2 Smart Link 与 Monitor Link

原理概述

在以太网络中，为了提高网络的可靠性，通常采用双归属上行方式进行组网，即一台交换机同时连接两台上行交换机，但是在二层网络中可能会带来环路问题。为了解决环路问题，可以采用 STP 技术，但 STP 的收敛时间较长，当主用链路故障时，将流量切换到

备用链路，只能是达到秒级的收敛速度，不适用于对收敛时间有很高要求的组网环境。

基于上述原因，华为公司针对双归属上行组网提出了 Smart Link 解决方案。网络中两条上行链路在正常情况下，只有一条处于连通状态，而另一条处于阻塞状态，从而防止了环路引起的广播风暴。当主用链路发生故障后，流量会在毫秒级的时间内迅速切换到备用链路上，保证了数据的正常转发。默认情况下，当原主用链路故障恢复时，将维持在阻塞状态，不进行抢占，从而保持网络稳定，可以手工配置回切功能使流量切换回原主用链路。Smart Link 配置简单，便于操作和维护。

Smart Link 虽然能够保证设备在本设备上行链路发生故障后快速进行倒换，但对于跨设备的链路故障不能提供有效保护，为此可以采用 Monitor Link。Monitor Link 用于扩展 Smart Link 的链路备份的范围，通过监控上游设备的上行链路，达到上行链路故障迅速传达给下游设备，从而触发 Smart Link 的主备链路切换，防止长时间因上行链路故障而出现网络中断，使 Smart Link 备份作用更为完善。

实验目的

- 理解 Smart Link 的应用场景
- 掌握 Smart Link 组的基本配置
- 掌握 Smart Link 回切功能的配置
- 掌握 Monitor Link 的基本配置

实验内容

本实验模拟公司网络场景。交换机 S4 作为公司出口设备连接外网，交换机 S1 是接入层交换机，负责员工终端接入，接入交换机通过两台交换机 S2 和 S3 双上行连接到 S4。针对此双上行组网，为了实现主备链路冗余备份及故障后的快速迁移，部署使用 Smart Link 技术，且为了进一步扩展 Smart Link 的备份范围，使用 Monitor Link 联动方式监控上游设备的上行链路来完善 Smart Link。

实验拓扑

Smart Link 与 Monitor Link 拓扑如图 5-2 所示。

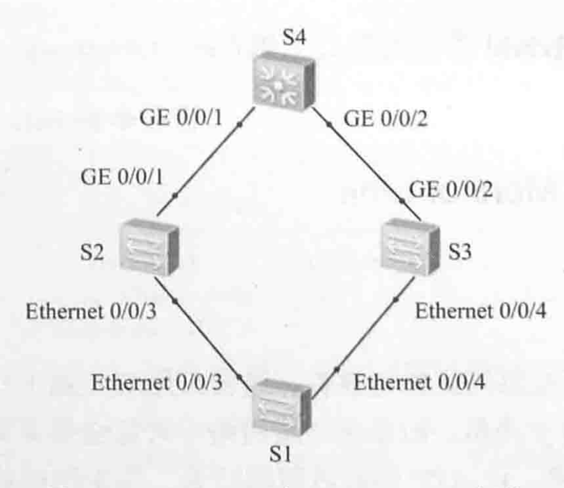


图 5-2 Smart Link 与 Monitor Link 拓扑

实验步骤

1. 配置 Smart Link

公司接入层交换机 S1 通过 S2 和 S3 双上行链路连接到出口交换机 S4，为了实现主备链路冗余备份及快速迁移，需要在 S1 上配置 Smart Link。

在 S1 上创建 Smart Link 组 1，并开启 Smart Link 组功能。

```
[S1]smart-link group 1
[S1-smlk-group1]smart-link enable
```

配置 Smart Link 时，需要在相关运行 Smart Link 的接口下关闭生成树协议。由于华为交换机默认开启了生成树协议，因此需要关闭 S1 交换机上 E 0/0/3 和 E 0/0/4 接口下的生成树协议。

```
[S1]interface Ethernet 0/0/3
[S1-Ethernet0/0/3]stp disable
[S1-Ethernet0/0/3]interface Ethernet 0/0/4
[S1-Ethernet0/0/4]stp disable
```

注意，如果相应接口下生成树协议未关闭，在配置 Smart Link 组功能时会报错，将会出现下面的提示信息。

```
Error: Adding a port failed. The port is already enabled with STP
```

进入到 Smart Link 组 1 下，配置 E 0/0/3 为主接口，E 0/0/4 为备份接口。

```
[S1]smart-link group 1
[S1-smlk-group1]port Ethernet 0/0/3 master
[S1-smlk-group1]port Ethernet 0/0/4 slave
```

配置完成后，使用 **display smart-link group 1** 命令查看主备状态。

```
[S1]display smart-link group 1
Smart Link group 1 information :
  Smart Link group was enabled
  There is no Load-Balance
  There is no protected-vlan reference-instance
  DeviceID: 4c1f-cc83-109c
  Member          Role      State       Flush     Count         Last-Flush-Time
 Ethernet0/0/3    Master    Active      0         0000/00/00    00:00:00 UTC+00:00
 Ethernet0/0/4    Slave     Inactive    0         0000/00/00    00:00:00 UTC+00:0
```

可以观察到，S1 交换机的 E 0/0/3 为主接口，且状态为 Active；E 0/0/4 为备份接口，状态为 Inactive。

2. 配置回切功能

当 S1 上主接口 E 0/0/3 出现故障关闭时，备份接口会立刻切换为 Active 状态。并且默认情况下，当原主接口恢复时，主接口不会自动回切到 Active 状态，需要手工配置回切功能。

将 S2 交换机 E 0/0/3 接口关闭，模拟故障发生，在 S1 上观察 Smart Link 组 1 的主备状态。

```
[S2]interface Ethernet 0/0/3
[S2-Ethernet0/0/3]shutdown

[S1]display smart-link group 1
Smart Link group 1 information :
  Smart Link group was enabled
```

```
    There is no Load-Balance
    There is no protected-vlan reference-instance
    DeviceID: 4c1f-cc83-109c
    Member         Role      State      Flush   Count        Last-Flush-Time
  Ethernet0/0/3    Master    Inactive   0       0000/00/00   00:00:00 UTC+00:00
  Ethernet0/0/4    Slave     Active     0       0000/00/00   00:00:00 UTC+00:00
```

可以观察到，S1 交换机 E 0/0/3 仍然为主接口，但是状态处于 Inactive，而 E 0/0/4 状态此时为 Active。

重新开启 S2 的 E 0/0/3 接口，再次在 S1 上观察 Smart Link 组 1 的主备状态。

```
[S2]interface Ethernet 0/0/3
[S2-Ethernet0/0/3]undo shutdown

[S1-Ethernet0/0/3]display smart-link group 1
Smart Link group 1 information :
  Smart Link group was enabled
  There is no Load-Balance
  There is no protected-vlan reference-instance
  DeviceID: 4c1f-cc83-109c
  Member         Role      State      Flush   Count        Last-Flush-Time
Ethernet0/0/3    Master    Inactive   0       0000/00/00   00:00:00 UTC+00:00
Ethernet0/0/4    Slave     Active     0       0000/00/00   00:00:00 UTC+00:00
```

可以观察到，接口的状态没有发生变化，E 0/0/3 接口仍然处于 Inactive 状态，并没有抢占原来的 Active 状态。即当主链路出现故障后，会自动切换到备份链路；而当原主链路故障恢复后，为了保持网络稳定，它将维持在阻塞状态，不进行抢占。如果需要原主链路恢复为 Active 状态，可以通过配置 Smart Link 组回切功能，在回切定时器超时后会自动切换到主链路。

在 S1 上使用 **restore enable** 命令开启回切功能，并将回切时间设置为 30s（默认为 60s）。

```
[S1]smart-link group 1
[S1-smlk-group1]restore enable
[S1-smlk-group1]timer wtr 30
```

等待 30s 后 S1 上会弹出如下信息，即已经产生了状态的切换。

```
Jun 26 2013 18:53:09-08:00 S1 %%01SMLK/4/SMLK_STATUS_LOG(l)[5]:The state of Smart link group 1 changed to MASTER.
```

查看 Smart Link 组 1 的主备状态。

```
[S1]display smart-link group 1
Smart Link group 1 information :
  Smart Link group was enabled
  Wtr-time is: 30 sec.
  There is no Load-Balance
  There is no protected-vlan reference-instance
  DeviceID: 4c1f-cc83-109c
  Member         Role      State      Flush   Count        Last-Flush-Time
Ethernet0/0/3    Master    Active     0       0000/00/00   00:00:00 UTC+00:00
Ethernet0/0/4    Slave     Inactive   0       0000/00/00   00:00:00 UTC+00:00
```

可以观察到，S1 的 E 0/0/3 接口状态又重新恢复到 Active 状态，而 E 0/0/4 接口回到了 Inactive 状态。

3. 配置 Monitor Link

Monitor Link 是对 Smart Link 进行补充而引入的接口联动方案，用于扩展 Smart Link

的链路备份的范围。通过监控上游设备的上行链路，而对下行链路进行同步设置，达到上游设备的上行链路故障迅速传达给下行设备，从而触发下游设备的 Smart Link 的主备链路切换，防止长时间因上行链路故障而出现网络故障。

正常情况下，S1 与 S2 之间的链路为主链路，但是当 S2 的上行接口 GE 0/0/1 故障时， Smart Link 无法感知故障，不会发生切换，导致网络中断。为了解决这一问题，需要在 S2 上配置 Monitor Link 监控上行接口，当 GE 0/0/1 故障时，使 S1 的 Smart Link 组切换。

为了模拟该场景，现将 S2 的 GE 0/0/1 接口关闭，并查看 Smart Link 组 1 的主备状态。

```
[S2]interface GigabitEthernet 0/0/1
[S2-GigabitEthernet0/0/1]shutdown

[S1]display smart-link group 1
Smart Link group 1 information :
  Smart Link group was enabled
  Wtr-time is: 30 sec.
  There is no Load-Balance
  There is no protected-vlan reference-instance
  DeviceID: 4c1f-cc83-109c
  Member          Role      State      Flush     Count          Last-Flush-Time
Ethernet0/0/3     Master    Active     0         0000/00/00     00:00:00 UTC+00:00
Ethernet0/0/4     Slave     Inactive   0         0000/00/00     00:00:00 UTC+00:00
```

可以观察到，当 S2 的上行 GE 0/0/1 接口出现故障以后，连接到下行链路的 S1 交换机无法感知到该故障，导致 S1 交换机的 Smart Link 无法进行切换，这样会导致连接到 S1 交换机仍然选择 E 0/0/3 接口转发数据，无法正常通信。

在 S2 上启用 Monitor Link 组 1，配置上行接口为 GE 0/0/1，下行接口为 E 0/0/3。

```
[S2]monitor-link group 1
[S2-mtlk-group1]port GigabitEthernet 0/0/1 uplink
[S2-mtlk-group1]port Ethernet 0/0/3 downlink
```

配置完成后，再次查看 S1 的 Smart Link 组 1 的主备状态。

```
[S1]display smart-link group 1
Smart Link group 1 information :
  Smart Link group was enabled
  Wtr-time is: 30 sec.
  There is no Load-Balance
  There is no protected-vlan reference-instance
  DeviceID: 4c1f-cc83-109c
  Member          Role      State      Flush     Count          Last-Flush-Time
Ethernet0/0/3     Master    Inactive   0         0000/00/00     00:00:00 UTC+00:00
 Ethernet0/0/4    Slave     Active     0         0000/00/00     00:00:00 UTC+00:00
```

观察发现 E 0/0/3 接口状态已经变为 Inactive，E 0/0/4 接口状态成为了 Active，流量已经被切换到 E 0/0/4 接口，保证了用户流量的正常转发。

修改 Monitor Link 组的回切时间为 10 秒（默认为 3s）。当 S2 的上行接口 GE 0/0/1 重新恢复以后，下行链路 Smart Link 组将在时间到期后，重新回切到主链路。

```
[S2-mtlk-group1]timer recover-time 10
```

重新开启 S2 的 GE 0/0/1 接口。

```
[S2]interface GigabitEthernet 0/0/1
[S2-GigabitEthernet0/0/1]undo shutdown
```

等待 40s 左右（加上步骤 2 中配置的 Smart Link 回切时间），查看 S1 的 Smart Link 组 1 的主备状态。

```
[S1]display smart-link group 1
Smart Link group 1 information :
  Smart Link group was enabled
  Wtr-time is: 30 sec.
  There is no Load-Balance
  There is no protected-vlan reference-instance
  DeviceID: 4c1f-cc83-109c
  Member          Role      State      Flush      Count          Last-Flush-Time
 Ethernet0/0/3    Master    Active     0          0000/00/00     00:00:00 UTC+00:00
 Ethernet0/0/4    Slave     Inactive   0          0000/00/00     00:00:00 UTC+00:00
```

可以观察到，此时 S1 的 E 0/0/3 接口重新恢复到了 Active 状态。

思考

Smart Link 和 Monitor Link 的联合使用可以确保链路出现故障后及时地切换，如果所有链路都正常，是否所有数据都只能通过主链路转发？

5.3 配置 Eth-Trunk 链路聚合

原理概述

在没有使用 Eth-Trunk 前，百兆以太网的双绞线在两个互连的网络设备间的带宽仅为 100Mbit/s。若想达到更高的数据传输速率，则需要更换传输媒介，使用千兆光纤或升级成为千兆以太网。这样的解决方案成本较高。如果采用 Eth-Trunk 技术把多个接口捆绑在一起，则可以以较低的成本满足提高接口带宽的需求。例如，把 3 个 100Mbit/s 的全双工接口捆绑在一起，就可以达到 300Mbit/s 的最大带宽。

Eth-Trunk 是一种捆绑技术，它将多个物理接口捆绑成一个逻辑接口，这个逻辑接口就称为 Eth-Trunk 接口，捆绑在一起的每个物理接口称为成员接口。Eth-Trunk 只能由以太网链路构成。Trunk 的优势在于：

■ 负载分担，在一个 Eth-Trunk 接口内，可以实现流量负载分担；

■ 提高可靠性，当某个成员接口连接的物理链路出现故障时，流量会切换到其他可用的链路上，从而提高整个 Trunk 链路的可靠性；

■ 增加带宽，Trunk 接口的总带宽是各成员接口带宽之和。

Eth-Trunk 在逻辑上把多条物理链路捆绑等同于一条逻辑链路，对上层数据透明传输。所有 Eth-Trunk 中物理接口的参数必须一致，Eth-Trunk 链路两端要求一致的物理参数有：Eth-Trunk 链路两端相连的物理接口类型、物理接口数量、物理接口的速率、物理接口的双工方式以及物理接口的流控方式。

实验目的

- 理解使用 Eth-Trunk 的应用场景

- 掌握配置 Eth-Trunk 链路聚合的方法（手工负载分担模式）
- 掌握配置 Eth-Trunk 链路聚合的方法（静态 LACP 模式）

实验内容

本实验模拟企业网络环境。S1 和 S2 为企业核心交换机，PC-1 属于 A 部门终端设备，PC-2 属于 B 部门终端设备。根据企业规划，S1 和 S2 之间线路原由一条光纤线路相连，但出于带宽和冗余角度考虑需要对其进行升级，可使用 Eth-Trunk 实现此需求。

实验拓扑

配置 Eth-Trunk 链路聚合的拓扑如图 5-3 所示。

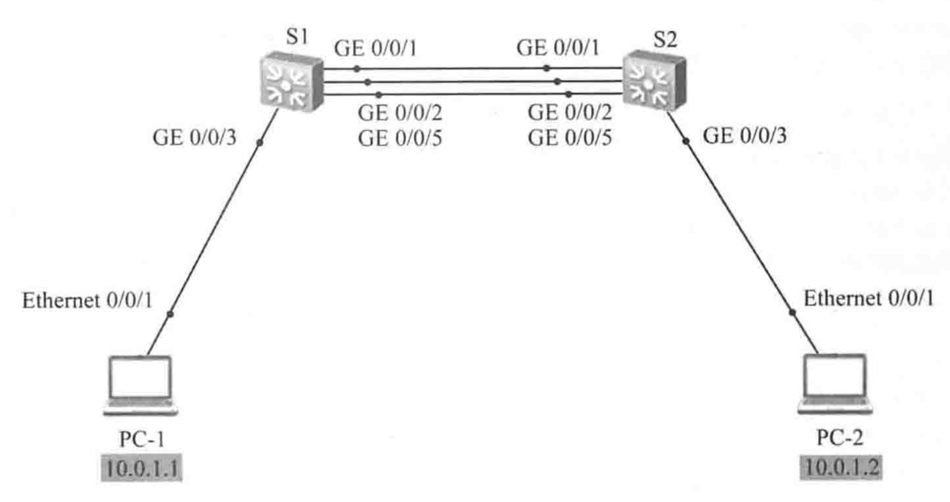


图 5-3 配置 Eth-Trunk 链路聚合拓扑

实验编址

实验编址见表 5-2。

表 5-2 实验编址

设备	接口	IP 地址	子网掩码	默认网关
PC-1	Ethernet 0/0/1	10.0.1.1	255.255.255.0	N/A
PC-2	Ethernet 0/0/1	10.0.1.2	255.255.255.0	N/A

MAC 地址

本实验的 MAC 地址见表 5-3。

表 5-3 MAC 地址

设备	全局 MAC 地址
S1（S5700）	4c1f-cc55-b90f
S2（S5700）	4c1f-cc71-68d4

实验步骤

1. 基本配置

根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测各 PC 之间的连通性。

```
PC>ping 10.0.1.2
Ping 10.0.1.2: 32 data bytes, Press Ctrl_C to break
From 10.0.1.2: bytes=32 seq=1 ttl=128 time=16 ms
From 10.0.1.2: bytes=32 seq=2 ttl=128 time=16 ms
From 10.0.1.2: bytes=32 seq=3 ttl=128 time<1 ms
From 10.0.1.2: bytes=32 seq=4 ttl=128 time=16 ms
From 10.0.1.2: bytes=32 seq=5 ttl=128 time=46 ms
--- 10.0.1.2 ping statistics ---
5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 0/18/46 ms
```

其余 PC 的连通性测试省略。

由于本实验场景需要，首先要将 S1 与 S2 上互连的 GE 0/0/2 和 GE 0/0/5 接口关闭。

```
[S1]interface GigabitEthernet 0/0/2
[S1-GigabitEthernet0/0/2]shutdown
[S1-GigabitEthernet0/0/2]interface GigabitEthernet 0/0/5
[S1-GigabitEthernet0/0/5]shutdown

[S2]interface GigabitEthernet 0/0/2
[S2-GigabitEthernet0/0/2]shutdown
[S2-GigabitEthernet0/0/2]interface GigabitEthernet 0/0/5
[S2-GigabitEthernet0/0/5]shutdown
```

2. 未配置 Eth-Trunk 时的现象验证

在原有的网络环境中，公司在两台核心交换机间只部署了一条链路。但随着业务增长，数据量的增大，带宽出现了瓶颈，已经无法满足公司的业务需求，也无法实现冗余备份。考虑到以上问题，公司网络管理员决定通过增加链路的方式来提升带宽。原链路只有一条，带宽为 1Gbit/s，在原有的网络基础上再增加一条链路，将带宽增加到 2Gbit/s。

模拟链路增加，开启 S1 和 S2 上的 GE 0/0/2 接口。

```
[S1]interface GigabitEthernet 0/0/2
[S1-GigabitEthernet0/0/2]undo shutdown

[S2]interface GigabitEthernet 0/0/2
[S2-GigabitEthernet0/0/2]undo shutdown
```

增加链路后，网络管理员考虑到，在该组网拓扑下，默认开启的 STP 协议一定会将其中一条链路阻塞掉。

查看 S1 和 S2 的 STP 状态信息。

```
[S1]display stp brief
 MSTID  Port                        Role  STP State     Protection
   0    GigabitEthernet0/0/1        DESI  FORWARDING      NONE
   0    GigabitEthernet0/0/2        DESI  FORWARDING      NONE
   0    GigabitEthernet0/0/3        DESI  FORWARDING      NONE

[S2]display stp brief
 MSTID  Port                        Role  STP State     Protection
```

```
    0     GigabitEthernet0/0/1        ROOT   FORWARDING        NONE
    0     GigabitEthernet0/0/2        ALTE   DISCARDING        NONE
    0     GigabitEthernet0/0/3        DESI   FORWARDING        NONE
```

可以观察到 S2 的 GE 0/0/2 接口处于丢弃状态。如果要实质性地增加 S1 和 S2 之间的带宽，显然单靠增加链路条数是不够的。生成树会阻塞多余接口，使得目前 S1 与 S2 之间的数据仍然仅通过 GE 0/0/1 接口传输。

3. 配置 Eth-Trunk 实现链路聚合（手工负载分担模式）

通过上一步骤，发现仅靠简单增加互连的链路，不但无法解决目前带宽不够用的问题，还会在切换时带来断网的问题，显然是不合理的。此时网络管理员通过配置 Eth-Trunk 链路聚合来增加链路带宽，并可确保冗余链路。

Eth-Trunk 工作模式可以分为两种：

■ 手工负载分担模式：需要手动创建链路聚合组，并配置多个接口加入到所创建的 Eth-Trunk 中；

■ 静态 LACP 模式：该模式通过 LACP 协议协商 Eth-Trunk 参数后自主选择活动接口。

在 S1 和 S2 上配置链路聚合，创建 Eth-Trunk 1 接口，并指定为手工负载分担模式。

```
[S1]interface Eth-Trunk 1
[S1-Eth-Trunk1]mode manual load-balance

[S2]interface Eth-Trunk 1
[S2-Eth-Trunk1]mode manual load-balance
```

将 S1 和 S2 的 GE 0/0/1 和 GE 0/0/2 分别加入到 Eth-Trunk 1 接口。

```
[S1]interface GigabitEthernet 0/0/1
[S1-GigabitEthernet0/0/1]eth-trunk 1
[S1-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2
[S1-GigabitEthernet0/0/2]eth-trunk 1

[S2]interface GigabitEthernet 0/0/1
[S2-GigabitEthernet0/0/1]eth-trunk 1
[S2-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2
[S2-GigabitEthernet0/0/2]eth-trunk 1
```

配置完成后，使用 **display eth-trunk 1** 命令查看 S1 和 S2 的 Eth-Trunk 1 接口状态。

```
[S1]display eth-trunk 1
Eth-Trunk1's state information is:
WorkingMode: NORMAL           Hash arithmetic: According to SIP-XOR-DIP
Least Active-linknumber: 1   Max Bandwidth-affected-linknumber: 8
Operate status: up            Number Of Up Port In Trunk: 2
--------------------------------------------------------------------------------
PortName                      Status      Weight
GigabitEthernet0/0/1          Up          1
GigabitEthernet0/0/2          Up          1

[S2]display eth-trunk 1
Eth-Trunk1's state information is:
WorkingMode: NORMAL           Hash arithmetic: According to SIP-XOR-DIP
Least Active-linknumber: 1   Max Bandwidth-affected-linknumber: 8
Operate status: up            Number Of Up Port In Trunk: 2
--------------------------------------------------------------------------------
PortName                      Status      Weight
```

```
GigabitEthernet0/0/1           Up           1
GigabitEthernet0/0/2           Up           1
```

可以观察到，S1 与 S2 的工作模式为 NORMAL（手工负载分担方式），GE 0/0/1 与 GE 0/0/2 接口已经添加到 Eth-Trunk 1 中，并且处于 UP 状态。

使用 **display interface eth-trunk 1** 命令查看 S2 的 Eth-Trunk 1 接口信息。

```
[S2]display interface Eth-Trunk 1
Eth-Trunk1 current state : UP
Line protocol current state : UP
Description:
Switch Port, PVID :    1, Hash arithmetic : According to SIP-XOR-DIP,Maximal BW: 4294967.29G, Current BW: 4294967.29G, The Maximum Frame Length is 9216
IP Sending Frames' Format is PKTFMT_ETHNT_2, Hardware address is 4c1f-cc71-68d4
Current system time: 2013-06-29 22:34:26-08:00
    Input bandwidth utilization  :    0%
    Output bandwidth utilization :    0%
-----------------------------------------------------------------------------
PortName                      Status      Weight
-----------------------------------------------------------------------------
GigabitEthernet0/0/1           UP           1
GigabitEthernet0/0/2           UP           1
-----------------------------------------------------------------------------
The Number of Ports in Trunk : 2
The Number of UP Ports in Trunk : 2
```

可以观察到，目前该接口的总带宽，是 GE 0/0/1 和 GE 0/0/2 接口带宽之和。

查看 S2 接口的生成树状态。

```
[S2]display stp brief
 MSTID  Port                    Role    STP State       Protection
   0    GigabitEthernet0/0/3    DESI    FORWARDING      NONE
   0    Eth-Trunk1              ROOT    FORWARDING      NONE
```

可以观察到，S2 的 2 个接口被捆绑成一个 Eth-Trunk 接口，并且该接口现在处于转发状态。

使用 **ping** 命令持续测试，同时将 S2 的 GE 0/0/1 或者 GE 0/0/2 接口关闭模拟故障发生。

```
PC>ping 10.0.1.2 -t
Ping 10.0.1.2: 32 data bytes, Press Ctrl_C to break
From 10.0.1.2: bytes=32 seq=1 ttl=128 time=63 ms
From 10.0.1.2: bytes=32 seq=2 ttl=128 time=31 ms
From 10.0.1.2: bytes=32 seq=3 ttl=128 time=31 ms
From 10.0.1.2: bytes=32 seq=4 ttl=128 time=16 ms
From 10.0.1.2: bytes=32 seq=5 ttl=128 time=31 ms
Request timeout!
From 10.0.1.2: bytes=32 seq=7 ttl=128 time=47 ms
From 10.0.1.2: bytes=32 seq=8 ttl=128 time=15 ms
From 10.0.1.2: bytes=32 seq=9 ttl=128 time=31 ms
From 10.0.1.2: bytes=32 seq=10 ttl=128 time=16 ms
```

可以观察到，当链路故障发生时，链路立刻进行切换，数据包仅丢了一个，并且只要物理链路有一条是正常的，Eth-Trunk 接口就不会断开，仍然可以保证数据的转发。可见，Eth-Trunk 在提高了带宽的情况下，也实现了链路冗余。模拟完成后将 S2 接口恢复。

4. 配置 Eth-Trunk 实现链路聚合（静态 LACP 模式）

在上一节中，假设两条链路中的一条出现了故障，只有一条链路正常工作的情况下

无法保证带宽。现网络管理员为公司再部署一条链路作为备份链路，并采用静态 LACP 模式配置 Eth-Trunk 实现两条链路同时转发，一条链路备份，当其中一条转发链路出现问题时，备份链路可立即进行数据转发。

开启 S1 与 S2 上的 GE 0/0/5 接口模拟增加了一条新链路。

```
[S1]interface GigabitEthernet 0/0/5
[S1-GigabitEthernet0/0/5]undo shutdown

[S2]interface GigabitEthernet 0/0/5
[S2-GigabitEthernet0/0/5]undo shutdown
```

在 S1 和 S2 上的 Eth-Trunk 1 接口下，将工作模式改为静态 LACP 模式。

```
[S1]interface Eth-Trunk 1
[S1-Eth-Trunk1]mode lacp-static
Error: Error in changing trunk working mode. There is(are) port(s) in the trunk.

[S2]interface Eth-Trunk 1
[S2-Eth-Trunk1]mode lacp-static
Error: Error in changing trunk working mode. There is(are) port(s) in the trunk.
```

发现报错，此时需要将先前已经加入到 Eth-Trunk 接口下的物理接口先删除。

```
[S1]interface GigabitEthernet 0/0/1
[S1-GigabitEthernet0/0/1]undo eth-trunk 1
[S1-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2
[S1-GigabitEthernet0/0/2]undo eth-trunk 1

[S2]interface GigabitEthernet 0/0/1
[S2-GigabitEthernet0/0/1]undo eth-trunk 1
[S2-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2
[S2-GigabitEthernet0/0/2]undo eth-trunk 1
```

删除完成后，再在 S1 和 S2 上的 Eth-Trunk 1 接口下，将工作模式改为静态 LACP 模式，并将 S1 和 S2 的 GE 0/0/1、GE 0/0/2 和 GE 0/0/5 接口分别加入到 Eth-Trunk 1 接口。

```
[S1]interface Eth-Trunk 1
[S1-Eth-Trunk1]mode lacp-static
[S1-Eth-Trunk1]interface GigabitEthernet 0/0/1
[S1-GigabitEthernet0/0/1]eth-trunk 1
[S1-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2
[S1-GigabitEthernet0/0/2]eth-trunk 1
[S1-GigabitEthernet0/0/2]interface GigabitEthernet 0/0/5
[S1-GigabitEthernet0/0/5]eth-trunk 1

[S2]interface Eth-Trunk 1
[S2-Eth-Trunk1]mode lacp-static
[S2-Eth-Trunk1]interface GigabitEthernet 0/0/1
[S2-GigabitEthernet0/0/1]eth-trunk 1
[S2-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2
[S2-GigabitEthernet0/0/2]eth-trunk 1
[S2-GigabitEthernet0/0/2]interface GigabitEthernet 0/0/5
[S2-GigabitEthernet0/0/5]eth-trunk 1
```

配置完成后，查看 S1 的 Eth-Trunk 1 接口状态。

```
[S1]display eth-trunk 1
Eth-Trunk1's state information is:
Local:
LAG ID: 1                   WorkingMode: STATIC
Preempt Delay: Disabled     Hash arithmetic: According to SIP-XOR-DIP
```

```
System Priority: 32768        System ID: 4c1f-cc55-b90f
Least Active-linknumber: 1   Max Active-linknumber: 8
Operate status: up            Number Of Up Port In Trunk: 3
--------------------------------------------------------------------------------
ActorPortName          Status   PortType PortPri PortNo PortKey PortState Weight
GigabitEthernet0/0/2   Selected 1000TG   32768   3      401     10111100  1
GigabitEthernet0/0/1   Selected 1000TG   32768   2      401     10111100  1
GigabitEthernet0/0/5   Selected 1000TG   32768   6      401     10111100  1
Partner:
--------------------------------------------------------------------------------
ActorPortName          SysPri   SystemID        PortPri PortNo PortKey PortState
GigabitEthernet0/0/2   32768    4c1f-cc71-68d4  32768   3      401     10111100
GigabitEthernet0/0/1   32768    4c1f-cc71-68d4  32768   2      401     10111100
GigabitEthernet0/0/5   32768    4c1f-cc71-68d4  32768   6      401     10111100
```

可以观察到，3 个接口默认都处于活动状态（Selected）。

将 S1 的系统优先级从默认的 32768 改为 100，使其成为主动端（值越低优先级越高），并按照主动端设备的接口来选择活动接口。两端设备选出主动端后，两端都会以主动端的接口优先级来选择活动接口。两端设备选择了一致的活动接口，活动链路组便可以建立起来，设置这些活动链路以负载分担的方式转发数据。

```
[S1]lacp priority 100
```

配置完成后，查看 S1 的 Eth-Trunk 1 接口状态。

```
[S1]display eth-trunk 1
Eth-Trunk1's state information is:
Local:
LAG ID: 1                     WorkingMode: STATIC
Preempt Delay: Disabled       Hash arithmetic: According to SIP-XOR-DIP
System Priority: 100          System ID: 4c1f-cc55-b90f
Least Active-linknumber: 1   Max Active-linknumber: 8
......
```

可以观察到，已经将 S1 的 LACP 系统优先级改为 100，而 S2 没修改，仍为默认值。

在 S1 上配置活动接口上限阈值为 2。

```
[S1]interface Eth-Trunk 1
[S1-Eth-Trunk1]max active-linknumber 2
```

在 S1 上配置接口的优先级确定活动链路。

```
[S1]interface GigabitEthernet 0/0/1
[S1-GigabitEthernet0/0/1]lacp priority 100
[S1-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2
[S1-GigabitEthernet0/0/2]lacp priority 100
```

配置接口的活动优先级将默认的 32768 改为 100，目的是使 GE 0/0/1 和 GE 0/0/2 接口成为活动状态。

配置完成后，查看 S1 的 Eth-Trunk 1 接口状态。

```
[S1]display eth-trunk 1
Eth-Trunk1's state information is:
Local:
LAG ID: 1                     WorkingMode: STATIC
Preempt Delay: Disabled       Hash arithmetic: According to SIP-XOR-DIP
System Priority: 100          System ID: 4c1f-cc55-b90f
Least Active-linknumber: 1   Max Active-linknumber: 2
Operate status: up            Number Of Up Port In Trunk: 2
--------------------------------------------------------------------------------
```

```
ActorPortName          Status    PortType PortPri PortNo PortKey PortState Weight
GigabitEthernet0/0/2   Selected 1000TG   1003          401      10111100    1
GigabitEthernet0/0/1   Selected 1000TG   1002          401      10111100    1
GigabitEthernet0/0/5   Unselect 1000TG   32768    6    401      10100000    1
Partner:
......
```

可以观察到，由于将接口的阈值改为2（默认活动接口最大阈值为8），该Eth-Trunk接口下将只有两个成员处于活动状态，并且具有负载分担能力。而GE 0/0/5接口已处于不活动状态（Unselect），该链路作为备份链路。当活动链路出现故障时，备份链路将会替代故障链路，保持数据传输的可靠性。

将S1的GE 0/0/1接口关闭，验证Eth-Trunk链路聚合信息。

```
[S1]interface GigabitEthernet 0/0/1
[S1-GigabitEthernet0/0/1]shutdown

[S1-GigabitEthernet0/0/1]display eth-trunk 1
Eth-Trunk1's state information is:
Local:
LAG ID: 1                    WorkingMode: STATIC
Preempt Delay: Disabled      Hash arithmetic: According to SIP-XOR-DIP
System Priority: 100         System ID: 4c1f-cc27-e139
Least Active-linknumber: 1   Max Active-linknumber: 2
Operate status: up           Number Of Up Port In Trunk: 2
--------------------------------------------------------------------------------
ActorPortName          Status     PortType PortPri PortNo PortKey PortState Weight
GigabitEthernet0/0/1   Unselect 1GE       100      2       305     10100010  1
GigabitEthernet0/0/2   Selected 1GE       100      3       305     10111100  1
GigabitEthernet0/0/5   Selected 1GE       32768    6       305     10111100  1
Partner:
......
```

可以观察到，S1的GE 0/0/1接口已经处于不活动状态，而GE 0/0/5接口为活动状态。如果将S1的GE 0/0/1接口开启后，又会恢复为活动状态，GE 0/0/5则为不活动状态。

至此，完成了整个Eth-Trunk的部署。

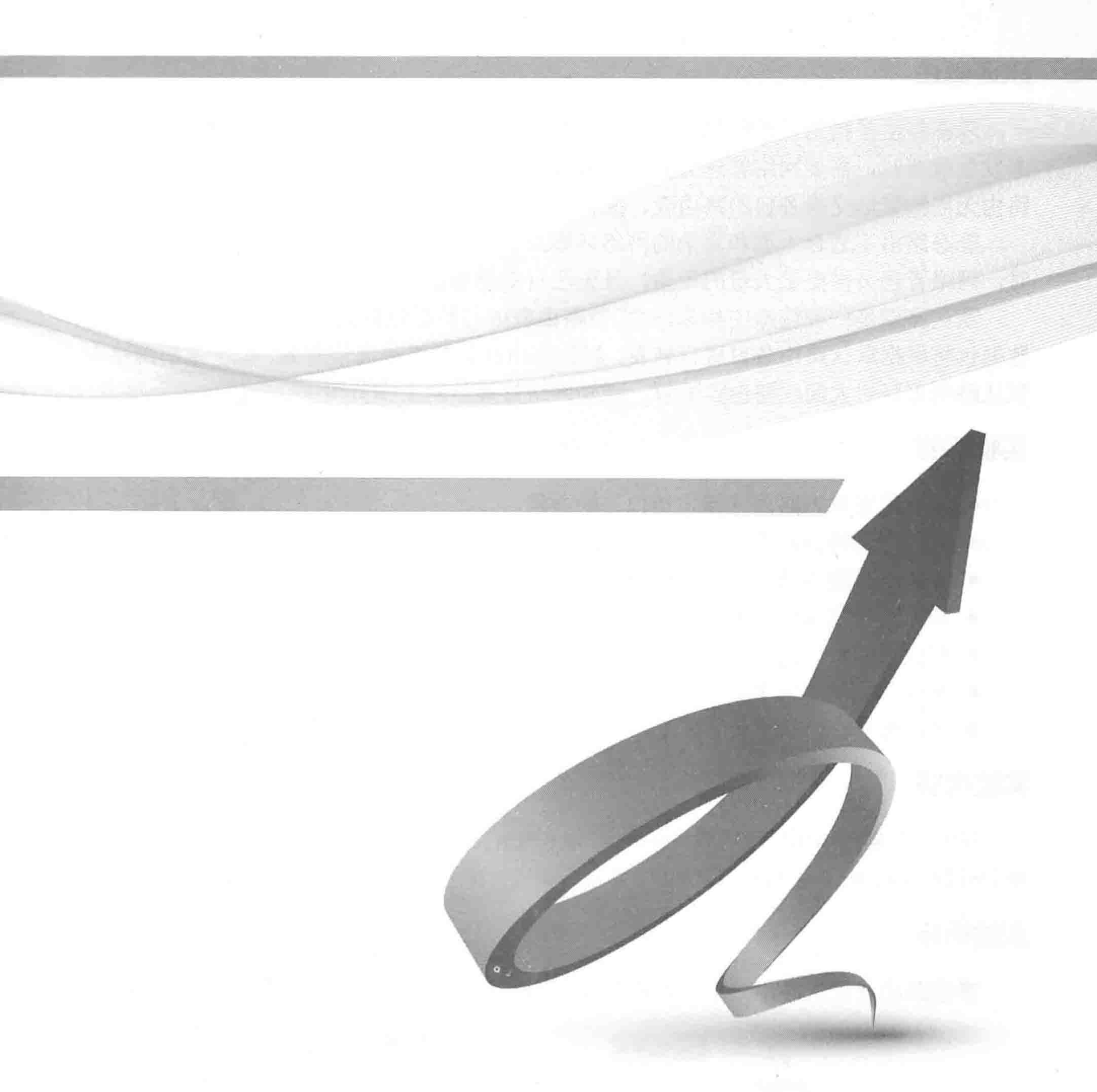
思考

当接口数超出最大负载阈值时，剩余接口是否转发流量？

第6章 静态路由

6.1 静态路由及默认路由基本配置

6.2 浮动静态路由及负载均衡



6.1 静态路由及默认路由基本配置

原理概述

静态路由是指用户或网络管理员手工配置的路由信息。当网络的拓扑结构或链路状态发生改变时，需要网络管理人员手工修改静态路由信息。相比于动态路由协议，静态路由无需频繁地交换各自的路由表，配置简单，比较适合小型、简单的网络环境。

静态路由不适合大型和复杂的网络环境，因为当网络拓扑结构和链路状态发生变化时，网络管理员需要做大量的调整，且无法自动感知错误发生，不易排错。

默认路由是一种特殊的静态路由，当路由表中与数据包目的地址没有匹配的表项时，数据包将根据默认路由条目进行转发。默认路由在某些时候非常有效，如在末梢网络中，默认路由可以大大简化路由器配置，减轻网络管理员的工作负担。

实验目的

- 掌握配置静态路由（指定接口）的方法
- 掌握配置静态路由（指定下一跳 IP 地址）的方法
- 掌握测试静态路由连通性的方法
- 掌握配置默认路由的方法
- 掌握测试默认路由的方法
- 掌握在简单网络中部署静态路由时的故障排除方法
- 掌握简单的网络优化方法

实验内容

在由 3 台路由器所组成的简单网络中，R1 与 R3 各自连接着一台主机，现在要求能够实现主机 PC-1 与 PC-2 之间的正常通信。本实验将通过配置基本的静态路由和默认路由来实现。

实验拓扑

静态路由及默认路由基本配置的拓扑如图 6-1 所示。

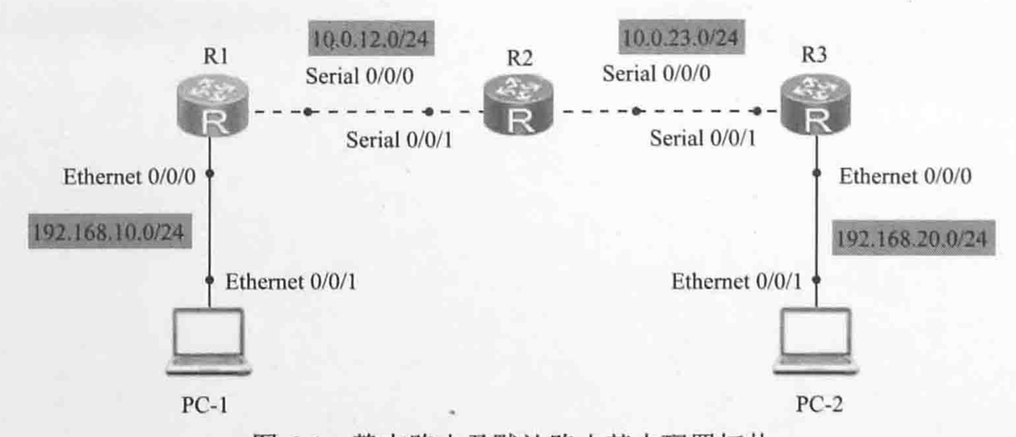


图 6-1 静态路由及默认路由基本配置拓扑

实验编址

实验编址见表 6-1。

表 6-1　　实验编址

设备	接口	IP 地址	子网掩码	默认网关
PC-1	Ethernet 0/0/1	192.168.10.10	255.255.255.0	192.168.10.1
R1（AR2220）	Ethernet 0/0/0	192.168.10.1	255.255.255.0	N/A
	Serial 0/0/0	10.0.12.1	255.255.255.0	N/A
R2（AR2220）	Serial 0/0/1	10.0.12.2	255.255.255.0	N/A
	Serial 0/0/0	10.0.23.2	255.255.255.0	N/A
R3（AR2220）	Serial 0/0/1	10.0.23.3	255.255.255.0	N/A
	Ethernet 0/0/0	192.168.20.3	255.255.255.0	N/A
PC-2	Ethernet 0/0/1	192.168.20.20	255.255.255.0	192.168.20.3

实验步骤

1. 基本配置

根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测各直连链路的连通性。

```
<R1>ping -c 1 192.168.10.10
  PING 192.168.10.10: 56   data bytes, press CTRL_C to break
    Reply from 192.168.10.10: bytes=56 Sequence=1 ttl=255 time=510 ms
  --- 192.168.10.10 ping statistics ---
    1 packet(s) transmitted
    1 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 510/510/510 ms
```

其余直连网段的连通性测试省略。

各直连链路间的 IP 连通性测试完成后，现尝试在主机 PC-1 上直接 **ping** 主机 PC-2。

```
PC>ping 192.168.20.20
Ping 192.168.20.20: 32 data bytes, Press Ctrl_C to break
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
……
```

发现无法连通，这时需要思考是什么问题导致了它们之间无法通信。

首先假设主机 PC-1 与 PC-2 之间如果能够正常连通，那么主机 A 将发送数据给其网关设备 R1；R1 收到后将根据数据包中的目的地址查看它的路由表，找到相应的目的网络的所在路由条目，并根据该条目中的下一跳和出接口信息将该数据转发给下一台路由器 R2；R2 采取同样的步骤将数据转发给 R3；最后 R3 也采取同样的步骤将数据转发给与自己直连的主机 PC-2；主机 PC-2 在收到数据后，与主机 PC-1 发送数据到 PC-2 的过程一样，再发送相应的回应消息给 PC-1。

在保证基本配置没有错误的情况下，首先查看主机 PC-1 与其网关设备 R1 间能否正常通信。

```
PC>ping 192.168.10.1
Ping 192.168.10.1: 32 data bytes, Press Ctrl_C to break
From 192.168.10.1: bytes=32 seq=1 ttl=255 time=16 ms
From 192.168.10.1: bytes=32 seq=2 ttl=255 time=16 ms
From 192.168.10.1: bytes=32 seq=3 ttl=255 time=16 ms
From 192.168.10.1: bytes=32 seq=4 ttl=255 time=31 ms
From 192.168.10.1: bytes=32 seq=5 ttl=255 time<1 ms
--- 192.168.10.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 0/15/31 ms
```

主机与网关之间通信正常，接下来检查网关设备 R1 上的路由表。

```
<R1>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7        Routes : 7
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      10.0.12.0/24  Direct  0    0           D   10.0.12.1       Serial0/0/0
      10.0.12.1/32  Direct  0    0           D   127.0.0.1       Serial0/0/0
      10.0.12.2/32  Direct  0    0           D   10.0.12.2       Serial0/0/0
      127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
   192.168.10.0/24  Direct  0    0           D   192.168.10.1    Ethernet0/0/0
   192.168.10.1/32  Direct  0    0           D   127.0.0.1       Ethernet0/0/0
```

可以看到在 R1 的路由表上，没有任何关于主机 PC-2 所在网段的信息。可以使用同样的方式查看 R2 与 R3 的路由表。

```
<R2>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8        Routes : 8
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      10.0.12.0/24  Direct  0    0           D   10.0.12.2       Serial0/0/1
      10.0.12.1/32  Direct  0    0           D   10.0.12.1       Serial0/0/1
      10.0.12.2/32  Direct  0    0           D   127.0.0.1       Serial0/0/1
      10.0.23.0/24  Direct  0    0           D   10.0.23.2       Serial0/0/0
      10.0.23.2/32  Direct  0    0           D   127.0.0.1       Serial0/0/0
      10.0.23.3/32  Direct  0    0           D   10.0.23.3       Serial0/0/0
      127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0

<R3>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7        Routes : 7
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      10.0.23.0/24  Direct  0    0           D   10.0.23.3       Serial0/0/1
      10.0.23.2/32  Direct  0    0           D   10.0.23.2       Serial0/0/1
      10.0.23.3/32  Direct  0    0           D   127.0.0.1       Serial0/0/1
      127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
```

```
        192.168.20.0/24  Direct  0    0        D   192.168.20.3    Ethernet0/0/0
        192.168.20.3/32  Direct  0    0        D   127.0.0.1       Ethernet0/0/0
```

可以看到在R2上没有任何关于主机PC-1和PC-2所在网段的信息，R3上没有任何关于主机PC-1所在网段的信息，验证了初始情况下各路由器的路由表上仅包括了与自身直接相连的网段的路由信息。

现在主机PC-1与PC-2之间跨越了若干个不同网段，要实现它们之间的通信，只通过简单的IP地址等基本配置是无法实现的，必须在3台路由器上添加相应的路由信息，可以通过配置静态路由来实现。

配置静态路由有两种方式，一种是在配置中采取指定下一跳IP地址的方式，另一种是指定出接口的方式。

2. 实现主机PC-1与PC-2之间的通信

在R1上配置目的网段为主机PC-2所在网段的静态路由，即目的IP地址为192.168.20.0，掩码为255.255.255.0。对于R1而言，要发送数据到主机PC-2，则必须先发送给R2，所以R2即为R1的下一跳路由器，R2与R1所在的直连链路上的物理接口的IP地址即为下一跳IP地址，即10.0.12.2。

```
[R1]ip route-static 192.168.20.0 255.255.255.0 10.0.12.2
```

配置完成后，查看R1上的路由表。

```
<R1>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8        Routes : 8
Destination/Mask    Proto    Pre   Cost      Flags NextHop         Interface
      10.0.12.0/24  Direct   0     0           D   10.0.12.1       Serial0/0/0
      10.0.12.1/32  Direct   0     0           D   127.0.0.1       Serial0/0/0
      10.0.12.2/32  Direct   0     0           D   10.0.12.2       Serial0/0/0
      127.0.0.0/8   Direct   0     0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct   0     0           D   127.0.0.1       InLoopBack0
   192.168.10.0/24  Direct   0     0           D   192.168.10.1    Ethernet0/0/0
   192.168.10.1/32  Direct   0     0           D   127.0.0.1       Ethernet0/0/0
   192.168.20.0/24  Static   60    0           RD  10.0.12.2       Serial0/0/0
```

配置完成后，可以在R1的路由表上查看到主机PC-2所在网段的路由信息。

采取同样的方式在R2上配置目的网段为主机PC-2所在网段的静态路由。

```
[R2]ip route-static 192.168.20.0 255.255.255.0 10.0.23.3
```

配置完成后，查看R2上的路由表。

```
<R2>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 9        Routes : 9
Destination/Mask    Proto    Pre   Cost      Flags NextHop         Interface
      10.0.12.0/24  Direct   0     0           D   10.0.12.2       Serial0/0/1
      10.0.12.1/32  Direct   0     0           D   10.0.12.1       Serial0/0/1
      10.0.12.2/32  Direct   0     0           D   127.0.0.1       Serial0/0/1
      10.0.23.0/24  Direct   0     0           D   10.0.23.2       Serial0/0/0
      10.0.23.2/32  Direct   0     0           D   127.0.0.1       Serial0/0/0
      10.0.23.3/32  Direct   0     0           D   10.0.23.3       Serial0/0/0
      127.0.0.0/8   Direct   0     0           D   127.0.0.1       InLoopBack0
```

```
        127.0.0.1/32   Direct   0    0      D    127.0.0.1       InLoopBack0
     192.168.20.0/24   Static   60   0      RD   10.0.23.3       Serial0/0/0
```

配置完成后，可以在 R2 的路由表上查看到主机 PC-2 所在网段的路由信息。

此时在主机 PC-1 上 **ping** 主机 PC-2。

```
PC>ping 192.168.20.20
Ping 192.168.20.20: 32 data bytes, Press Ctrl_C to break
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
……
```

发现仍然无法连通。在主机 PC-1 的 E0/0/1 接口上进行数据抓包，可以观察到如图 6-2 所示的现象。

No.	Time	Source	Destination	Protocol	Info
1	0.000000	HuaweiTe_cf:ca:36	Broadcast	ARP	Who has 192.168.10.1? Tell 192.168.10.10
2	0.015000	HuaweiTe_06:35:f3	HuaweiTe_cf:ca:36	ARP	192.168.10.1 is at 00:e0:fc:06:35:f3
3	0.015000	192.168.10.10	192.168.20.20	ICMP	Echo (ping) request (id=0xebde, seq(be/le)=1/256, ttl=128)
4	2.015000	192.168.10.10	192.168.20.20	ICMP	Echo (ping) request (id=0xedde, seq(be/le)=2/512, ttl=128)
5	4.015000	192.168.10.10	192.168.20.20	ICMP	Echo (ping) request (id=0xefde, seq(be/le)=3/768, ttl=128)
6	6.015000	192.168.10.10	192.168.20.20	ICMP	Echo (ping) request (id=0xf1de, seq(be/le)=4/1024, ttl=128)
7	8.015000	192.168.10.10	192.168.20.20	ICMP	Echo (ping) request (id=0xf3de, seq(be/le)=5/1280, ttl=128)

图 6-2 抓包观察

此时主机 PC-1 仅发送了 ICMP 请求消息，并没有收到任何回应消息。原因在于现在仅仅实现了 PC-1 能够通过路由将数据正常转发给 PC-2，而 PC-2 仍然无法发送数据给 PC-1，所以同样需要在 R2 和 R3 的路由表上添加 PC-1 所在网段的路由信息。

在 R3 上配置目的网段为 PC-1 所在网段的静态路由，即目的 IP 地址为 192.168.10.0，目的地址的掩码除了可以采用点分十进制的格式表示外，还可以直接使用掩码长度，即 24 来表示。对于 R3 而言，要发送数据到 PC-1，则必须先发送给 R2，所以 R3 与 R2 所在直连链路上的物理接口 S 0/0/1 即为数据转发接口，也称为出接口，在配置中指定该接口即可。

```
[R3]ip route-static 192.168.10.0 24 Serial 0/0/1
```

采取同样的方式在 R2 上配置目的网段为 PC-1 所在网段的静态路由。

```
[R2]ip route-static 192.168.10.0 24 Serial 0/0/1
```

配置完成后，查看 R1、R2、R3 上的路由表。

```
<R1>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8        Routes : 8
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
        10.0.12.0/24   Direct   0    0         D   10.0.12.1       Serial0/0/0
        10.0.12.1/32   Direct   0    0         D   127.0.0.1       Serial0/0/0
        10.0.12.2/32   Direct   0    0         D   10.0.12.2       Serial0/0/0
        127.0.0.0/8    Direct   0    0         D   127.0.0.1       InLoopBack0
        127.0.0.1/32   Direct   0    0         D   127.0.0.1       InLoopBack0
     192.168.10.0/24   Direct   0    0         D   192.168.10.1    Ethernet0/0/0
     192.168.10.1/32   Direct   0    0         D   127.0.0.1       Ethernet0/0/0
     192.168.20.0/24   Static   60   0         RD  10.0.12.2       Serial0/0/0
```

```
<R2>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 10        Routes : 10
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      10.0.12.0/24  Direct  0    0           D   10.0.12.2       Serial0/0/1
      10.0.12.1/32  Direct  0    0           D   10.0.12.1       Serial0/0/1
      10.0.12.2/32  Direct  0    0           D   127.0.0.1       Serial0/0/1
      10.0.23.0/24  Direct  0    0           D   10.0.23.2       Serial0/0/0
      10.0.23.2/32  Direct  0    0           D   127.0.0.1       Serial0/0/0
      10.0.23.3/32  Direct  0    0           D   10.0.23.3       Serial0/0/0
      127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
   192.168.10.0/24  Static  60   0           D   10.0.12.1       Serial0/0/1
   192.168.20.0/24  Static  60   0           RD  10.0.23.3       Serial0/0/0

<R3>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8         Routes : 8
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      10.0.23.0/24  Direct  0    0           D   10.0.23.2       Serial0/0/1
      10.0.23.2/32  Direct  0    0           D   10.0.23.2       Serial0/0/1
      10.0.23.3/32  Direct  0    0           D   127.0.0.1       Serial0/0/1
      127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
   192.168.10.0/24  Static  60   0           D   10.0.23.2       Serial0/0/1
   192.168.20.0/24  Direct  0    0           D   192.168.20.3    Ethernet0/0/0
   192.168.20.3/32  Direct  0    0           D   127.0.0.1       Ethernet0/0/0
```

可以看到，现在每台路由器上都拥有了主机 PC-1 与 PC-2 所在网段的路由信息。再在主机 PC-1 上 **ping** 主机 PC-2。

```
PC>ping 192.168.20.20
Ping 192.168.20.20: 32 data bytes, Press Ctrl_C to break
From 192.168.20.20: bytes=32 seq=1 ttl=125 time=78 ms
From 192.168.20.20: bytes=32 seq=2 ttl=125 time=47 ms
From 192.168.20.20: bytes=32 seq=3 ttl=125 time=47 ms
From 192.168.20.20: bytes=32 seq=4 ttl=125 time=62 ms
From 192.168.20.20: bytes=32 seq=5 ttl=125 time=63 ms
--- 192.168.20.20 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
0.00% packet loss
  round-trip min/avg/max = 47/59/78 ms
```

可以连通，即现在已经实现了主机 PC-1 与 PC-2 之间的正常通信。

3. 实现全网全通来增强网络的可靠性

经过上面的步骤，主机 PC-1 与 PC-2 之间已经能够正常通信。假设此时网络突然出现故障，主机 PC-1 侧的网络管理员发现无法与 PC-2 正常通信，于是先测试与网关设备 R1 间的连通性。

```
PC>ping 192.168.10.1
```

```
Ping 192.168.10.1: 32 data bytes, Press Ctrl_C to break
From 192.168.10.1: bytes=32 seq=1 ttl=255 time<1 ms
From 192.168.10.1: bytes=32 seq=2 ttl=255 time=16 ms
From 192.168.10.1: bytes=32 seq=3 ttl=255 time=16 ms
From 192.168.10.1: bytes=32 seq=4 ttl=255 time=16 ms
From 192.168.10.1: bytes=32 seq=5 ttl=255 time=16 ms
--- 192.168.10.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
0.00% packet loss
  round-trip min/avg/max = 0/12/16 ms
```

发现与网关间的通信正常，再测试与主机 PC-2 的网关设备 R3 间的连通性。

```
PC>ping 10.0.23.3
Ping 10.0.23.3: 32 data bytes, Press Ctrl_C to break
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
……
```

发现无法与 R3 正常通信，这也意味着此时网络管理员将无法通过主机 PC-1 登录到 R3 上进一步排除故障，由此可见，保证全网的连通性能够增强整网的可靠性，提高网络的可维护性及健壮性。

因此有必要在 R1 的路由表中添加 R2 与 R3 间直连网段的路由信息，同样也应在 R3 的路由表中添加 R1 与 R2 间直连网段的路由信息，实现全网全通。

```
[R1]ip route-static 10.0.23.0 24 10.0.12.2

[R3]ip route-static 10.0.12.0 24 Serial 0/0/1
```

配置完成后，查看 R1、R2、R3 的路由表，注意观察新增的条目。

```
<R1>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 9        Routes : 9
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      10.0.12.0/24  Direct  0    0           D   10.0.12.1       Serial0/0/0
      10.0.12.1/32  Direct  0    0           D   127.0.0.1       Serial0/0/0
      10.0.12.2/32  Direct  0    0           D   10.0.12.2       Serial0/0/0
      10.0.23.0/24  Static  60   0           RD  10.0.12.2       Serial0/0/0
       127.0.0.0/8  Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
   192.168.10.0/24  Direct  0    0           D   192.168.10.1    Ethernet0/0/0
   192.168.10.1/32  Direct  0    0           D   127.0.0.1       Ethernet0/0/0
   192.168.20.0/24  Static  60   0           RD  10.0.12.2       Serial0/0/0

<R2>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 10       Routes : 10
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      10.0.12.0/24  Direct  0    0           D   10.0.12.2       Serial0/0/1
```

```
      10.0.12.1/32   Direct  0    0          D    10.0.12.1       Serial0/0/1
      10.0.12.2/32   Direct  0    0          D    127.0.0.1       Serial0/0/1
      10.0.23.0/24   Direct  0    0          D    10.0.23.2       Serial0/0/0
      10.0.23.2/32   Direct  0    0          D    127.0.0.1       Serial0/0/0
      10.0.23.3/32   Direct  0    0          D    10.0.23.3       Serial0/0/0
      127.0.0.0/8    Direct  0    0          D    127.0.0.1       InLoopBack0
      127.0.0.1/32   Direct  0    0          D    127.0.0.1       InLoopBack0
      192.168.10.0/24  Static  60   0        D    10.0.12.1       Serial0/0/1
      192.168.20.0/24  Static  60   0        RD   10.0.23.3       Serial0/0/0

<R3>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 9        Routes : 9
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      10.0.12.0/24   Static  60   0          D    10.0.23.2       Serial0/0/1
      10.0.23.0/24   Direct  0    0          D    10.0.23.3       Serial0/0/1
      10.0.23.2/32   Direct  0    0          D    10.0.23.2       Serial0/0/1
      10.0.23.3/32   Direct  0    0          D    127.0.0.1       Serial0/0/1
      127.0.0.0/8    Direct  0    0          D    127.0.0.1       InLoopBack0
      127.0.0.1/32   Direct  0    0          D    127.0.0.1       InLoopBack0
   192.168.10.0/24   Static  60   0          D    10.0.23.2       Serial0/0/1
   192.168.20.0/24   Direct  0    0          D    192.168.20.3    Ethernet0/0/0
   192.168.20.3/32   Direct  0    0          D    127.0.0.1       Ethernet0/0/0
```

此时再在主机 PC-1 上测试与 R3 间的连通性。

```
PC>ping 10.0.23.3
Ping 10.0.23.3: 32 data bytes, Press Ctrl_C to break
From 10.0.23.3: bytes=32 seq=1 ttl=253 time=47 ms
From 10.0.23.3: bytes=32 seq=2 ttl=253 time=47 ms
From 10.0.23.3: bytes=32 seq=3 ttl=253 time=47 ms
From 10.0.23.3: bytes=32 seq=4 ttl=253 time=47 ms
From 10.0.23.3: bytes=32 seq=5 ttl=253 time=93 ms
--- 10.0.23.3 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
0.00% packet loss
  round-trip min/avg/max = 47/56/93 ms
```

测试成功，主机 PC-1 可以顺利与 R3 通信，同样主机 PC-2 此时也能够与 R1 进行通信，测试过程这里省略。

4. 使用默认路由实现简单的网络优化

通过适当减少设备上的配置工作量，能够帮助网络管理员在进行故障排除时更轻松地定位故障，且相对较少的配置量也能减少在配置时出错的可能，另一方面，也能够相对减少对设备本身硬件的负担。

默认路由是一种特殊的静态路由，使用默认路由可以简化路由器上的配置。

查看此时 R1 上的路由表。

```
<R1>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 9        Routes : 9
```

```
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      10.0.12.0/24  Direct  0    0           D   10.0.12.1       Serial0/0/0
      10.0.12.1/32  Direct  0    0           D   127.0.0.1       Serial0/0/0
      10.0.12.2/32  Direct  0    0           D   10.0.12.2       Serial0/0/0
      10.0.23.0/24  Static  60   0           RD  10.0.12.2       Serial0/0/0
      127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
   192.168.10.0/24  Direct  0    0           D   192.168.10.1    Ethernet0/0/0
   192.168.10.1/32  Direct  0    0           D   127.0.0.1       Ethernet0/0/0
   192.168.20.0/24  Static  60   0           RD  10.0.12.2       Serial0/0/0
```

此时 R1 上存在两条先前经过手动配置的静态路由条目，且它们的下一跳和出接口都一致。

现在在 R1 上配置一条默认路由，即目的网段和掩码为全 0，表示任何网络，下一跳为 10.0.12.2，并删除先前配置的两条静态路由。

```
[R1]ip route-static 0.0.0.0 0 10.0.12.2
[R1]undo ip route-static 10.0.23.0 255.255.255.0 10.0.12.2
[R1]undo ip route-static 192.168.20.0 255.255.255.0 10.0.12.2
```

配置完成后，查看 R1 的路由表。

```
<R1>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8          Routes : 8
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      0.0.0.0/0     Static  60   0           RD  10.0.12.2       Serial0/0/0
      10.0.12.0/24  Direct  0    0           D   10.0.12.1       Serial0/0/0
      10.0.12.1/32  Direct  0    0           D   127.0.0.1       Serial0/0/0
      10.0.12.2/32  Direct  0    0           D   10.0.12.2       Serial0/0/0
      127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
      192.168.10.0/24 Direct 0   0           D   192.168.10.1    Ethernet0/0/0
      192.168.10.1/32 Direct 0   0           D   127.0.0.1       Ethernet0/0/0
```

再测试主机 PC-1 与 PC-2 间的通信。

```
PC>ping 192.168.20.20
Ping 192.168.20.20: 32 data bytes, Press Ctrl_C to break
From 192.168.20.20: bytes=32 seq=1 ttl=125 time=63 ms
From 192.168.20.20: bytes=32 seq=2 ttl=125 time=47 ms
From 192.168.20.20: bytes=32 seq=3 ttl=125 time=31 ms
From 192.168.20.20: bytes=32 seq=4 ttl=125 time=47 ms
From 192.168.20.20: bytes=32 seq=5 ttl=125 time=47 ms
--- 192.168.20.20 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
0.00% packet loss
  round-trip min/avg/max = 31/47/63 ms
```

发现主机 PC-1 与 PC-2 间的通信正常，证明使用默认路由不但能够实现与静态路由同样的效果，而且还能够减少配置量。在 R3 上可以进行同样的配置。

```
[R3]ip route-static 0.0.0.0 0 Serial 0/0/1
[R3]undo ip route-static 10.0.12.0 255.255.255.0 Serial 0/0/1
[R3]undo ip route-static 192.168.10.0 255.255.255.0 Serial 0/0/1
```

再次测试主机 PC-1 与 PC-2 间的通信。

```
PC>ping 192.168.20.20
Ping 192.168.20.20: 32 data bytes, Press Ctrl_C to break
From 192.168.20.20: bytes=32 seq=1 ttl=125 time=78 ms
From 192.168.20.20: bytes=32 seq=2 ttl=125 time=62 ms
From 192.168.20.20: bytes=32 seq=3 ttl=125 time=47 ms
From 192.168.20.20: bytes=32 seq=4 ttl=125 time=78 ms
From 192.168.20.20: bytes=32 seq=5 ttl=125 time=62 ms
--- 192.168.20.20 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
0.00% packet loss
  round-trip min/avg/max = 47/65/78 ms
```

主机 PC-1 与 PC-2 间的通信正常。

提示　在配置过程中，顺序是先配置默认路由，再删除原有的静态路由配置，这样的操作可以避免网络出现通信中断，即要在配置过程当中注意操作的规范性与合理性。

思考

在静态路由配置当中，可以采取指定下一跳 IP 地址的方式，也可以采取指定出接口的方式，这两种方式存在着什么区别？

6.2　浮动静态路由及负载均衡

原理概述

浮动静态路由（Floating Static Route）是一种特殊的静态路由，通过配置去往相同的目的网段，但优先级不同的静态路由，以保证在网络中优先级较高的路由，即主路由失效的情况下，提供备份路由。正常情况下，备份路由不会出现在路由表中。

负载均衡（Load sharing），当数据有多条可选路径前往同一目的网络，可以通过配置相同优先级和开销的静态路由实现负载均衡，使得数据的传输均衡地分配到多条路径上，从而实现数据分流、减轻单条路径负载过重的效果。而当其中某一条路径失效时，其他路径仍然能够正常传输数据，也起到了冗余作用。

实验目的

- 理解浮动静态路由的应用场景
- 掌握配置浮动静态路由的方法
- 掌握测试浮动静态路由的方法
- 掌握配置静态路由负载均衡的方法
- 掌握测试静态路由负载均衡的方法

实验内容

R2 为某公司总部，R1 与 R3 是两个分部，主机 PC-1 与 PC-2 所在的网段分别模拟

```
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      10.0.12.0/24  Direct  0    0           D   10.0.12.1       Serial0/0/0
      10.0.12.1/32  Direct  0    0           D   127.0.0.1       Serial0/0/0
      10.0.12.2/32  Direct  0    0           D   10.0.12.2       Serial0/0/0
      10.0.23.0/24  Static  60   0           RD  10.0.12.2       Serial0/0/0
       127.0.0.0/8  Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
   192.168.10.0/24  Direct  0    0           D   192.168.10.1    Ethernet0/0/0
   192.168.10.1/32  Direct  0    0           D   127.0.0.1       Ethernet0/0/0
   192.168.20.0/24  Static  60   0           RD  10.0.12.2       Serial0/0/0
```

此时 R1 上存在两条先前经过手动配置的静态路由条目，且它们的下一跳和出接口都一致。

现在在 R1 上配置一条默认路由，即目的网段和掩码为全 0，表示任何网络，下一跳为 10.0.12.2，并删除先前配置的两条静态路由。

```
[R1]ip route-static 0.0.0.0 0 10.0.12.2
[R1]undo ip route-static 10.0.23.0 255.255.255.0 10.0.12.2
[R1]undo ip route-static 192.168.20.0 255.255.255.0 10.0.12.2
```

配置完成后，查看 R1 的路由表。

```
<R1>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8          Routes : 8
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
        0.0.0.0/0   Static  60   0           RD  10.0.12.2       Serial0/0/0
        10.0.12.0/24  Direct  0    0           D   10.0.12.1       Serial0/0/0
        10.0.12.1/32  Direct  0    0           D   127.0.0.1       Serial0/0/0
        10.0.12.2/32  Direct  0    0           D   10.0.12.2       Serial0/0/0
        127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
        127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
        192.168.10.0/24  Direct  0    0           D   192.168.10.1    Ethernet0/0/0
        192.168.10.1/32  Direct  0    0           D   127.0.0.1       Ethernet0/0/0
```

再测试主机 PC-1 与 PC-2 间的通信。

```
PC>ping 192.168.20.20
Ping 192.168.20.20: 32 data bytes, Press Ctrl_C to break
From 192.168.20.20: bytes=32 seq=1 ttl=125 time=63 ms
From 192.168.20.20: bytes=32 seq=2 ttl=125 time=47 ms
From 192.168.20.20: bytes=32 seq=3 ttl=125 time=31 ms
From 192.168.20.20: bytes=32 seq=4 ttl=125 time=47 ms
From 192.168.20.20: bytes=32 seq=5 ttl=125 time=47 ms
--- 192.168.20.20 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
0.00% packet loss
  round-trip min/avg/max = 31/47/63 ms
```

发现主机 PC-1 与 PC-2 间的通信正常，证明使用默认路由不但能够实现与静态路由同样的效果，而且还能够减少配置量。在 R3 上可以进行同样的配置。

```
[R3]ip route-static 0.0.0.0 0 Serial 0/0/1
[R3]undo ip route-static 10.0.12.0 255.255.255.0 Serial 0/0/1
[R3]undo ip route-static 192.168.10.0 255.255.255.0 Serial 0/0/1
```

再次测试主机 PC-1 与 PC-2 间的通信。

```
PC>ping 192.168.20.20
Ping 192.168.20.20: 32 data bytes, Press Ctrl_C to break
From 192.168.20.20: bytes=32 seq=1 ttl=125 time=78 ms
From 192.168.20.20: bytes=32 seq=2 ttl=125 time=62 ms
From 192.168.20.20: bytes=32 seq=3 ttl=125 time=47 ms
From 192.168.20.20: bytes=32 seq=4 ttl=125 time=78 ms
From 192.168.20.20: bytes=32 seq=5 ttl=125 time=62 ms
--- 192.168.20.20 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
0.00% packet loss
  round-trip min/avg/max = 47/65/78 ms
```

主机 PC-1 与 PC-2 间的通信正常。

提示 在配置过程中，顺序是先配置默认路由，再删除原有的静态路由配置，这样的操作可以避免网络出现通信中断，即要在配置过程当中注意操作的规范性与合理性。

思考

在静态路由配置当中，可以采取指定下一跳 IP 地址的方式，也可以采取指定出接口的方式，这两种方式存在着什么区别？

6.2　浮动静态路由及负载均衡

原理概述

浮动静态路由（Floating Static Route）是一种特殊的静态路由，通过配置去往相同的目的网段，但优先级不同的静态路由，以保证在网络中优先级较高的路由，即主路由失效的情况下，提供备份路由。正常情况下，备份路由不会出现在路由表中。

负载均衡（Load sharing），当数据有多条可选路径前往同一目的网络，可以通过配置相同优先级和开销的静态路由实现负载均衡，使得数据的传输均衡地分配到多条路径上，从而实现数据分流、减轻单条路径负载过重的效果。而当其中某一条路径失效时，其他路径仍然能够正常传输数据，也起到了冗余作用。

实验目的

- 理解浮动静态路由的应用场景
- 掌握配置浮动静态路由的方法
- 掌握测试浮动静态路由的方法
- 掌握配置静态路由负载均衡的方法
- 掌握测试静态路由负载均衡的方法

实验内容

R2 为某公司总部，R1 与 R3 是两个分部，主机 PC-1 与 PC-2 所在的网段分别模拟

两个分部中的办公网络。现需要总部与各个分部、分部与分部之间都能够通信，且分部之间在通信时，之间的直连链路为主用链路，通过总部的链路为备用链路。本实验使用浮动静态路由实现需求，并再根据实际需求实现负载均衡来优化网络。

实验拓扑

浮动静态路由及负载均衡的拓扑如图 6-3 所示。

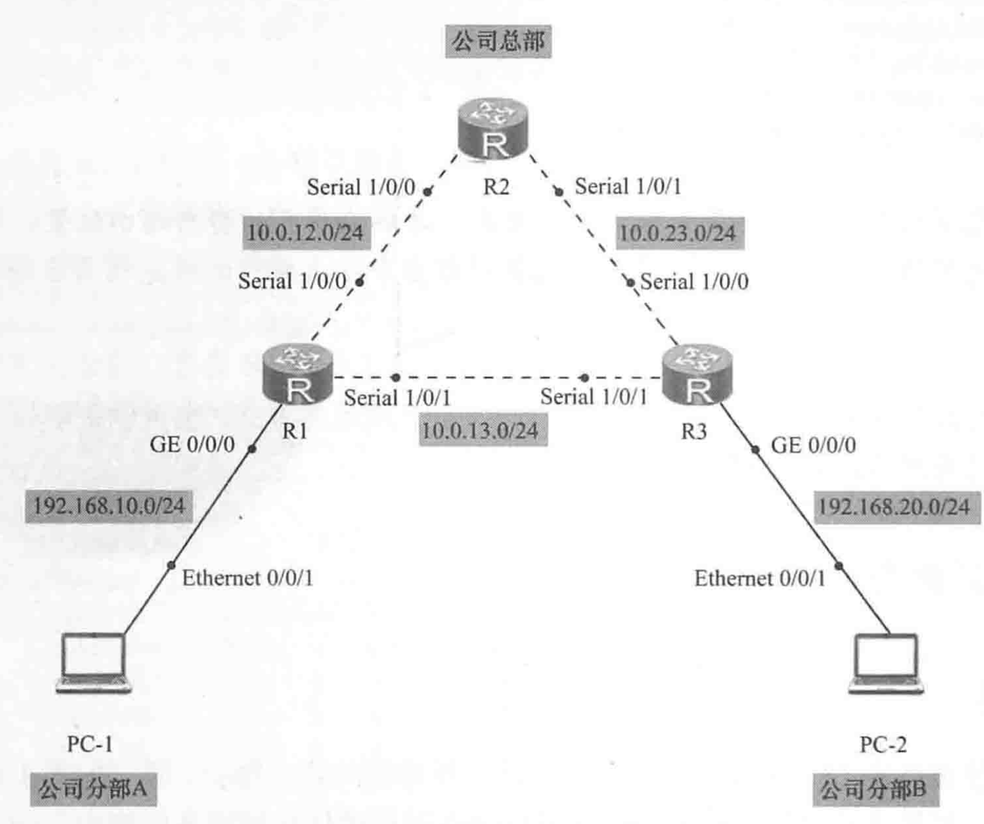


图 6-3 浮动静态路由及负载均衡拓扑

实验编址

实验编址见表 6-2。

表 6-2 实验编址

设备	接口	IP 地址	子网掩码	默认网关
PC-1	Ethernet 0/0/1	192.168.10.10	255.255.255.0	192.168.10.1
R1（AR2220）	GE 0/0/0	192.168.10.1	255.255.255.0	N/A
	Serial 1/0/0	10.0.12.1	255.255.255.0	N/A
	Serial 1/0/1	10.0.13.1	255.255.255.0	N/A
R2（AR2220）	Serial 1/0/0	10.0.12.2	255.255.255.0	N/A
	Serial 1/0/1	10.0.23.2	255.255.255.0	N/A
R3（AR2220）	Serial 1/0/0	10.0.23.3	255.255.255.0	N/A
	Serial 1/0/1	10.0.13.3	255.255.255.0	N/A
	GE 0/0/0	192.168.20.1	255.255.255.0	N/A
PC-2	Ethernet 0/0/1	192.168.20.20	255.255.255.0	192.168.20.1

实验步骤

1. 基本配置

根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测各直连链路的连通性。

```
<R1>ping -c 1 10.0.12.2
  PING 10.1.12.2: 56   data bytes, press CTRL_C to break
    Reply from 10.0.12.2: bytes=56 Sequence=1 ttl=255 time=510 ms
  --- 10.0.12.2 ping statistics ---
    1 packet(s) transmitted
    1 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 510/510/510 ms
```

其余直连网段的连通性测试省略。

2. 实现两分部间、总部与两分部间的通信

在 R1 上配置目的网段为主机 PC-2 所在网段的静态路由，在 R3 上配置目的网段为主机 PC-1 所在网段的静态路由，在 R2 上配置目的网段分别为主机 PC-1 和 PC-2 所在网段的静态路由。

```
[R1]ip route-static 192.168.20.0 24 10.0.13.3

[R2]ip route-static 192.168.20.0 24 10.0.23.3
[R2]ip route-static 192.168.10.0 24 10.0.12.1

[R3]ip route-static 192.168.10.0 24 10.0.13.1
```

配置完成后，在 R1 上查看路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8        Routes : 8
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      10.0.13.0/24  Direct  0    0           D   10.0.13.1       Serial1/0/1
      10.0.13.1/32  Direct  0    0           D   127.0.0.1       Serial1/0/1
      10.0.13.3/32  Direct  0    0           D   10.0.13.3       Serial1/0/1
      127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
   192.168.10.0/24  Direct  0    0           D   192.168.10.1    GigabitEthernet0/0/0
   192.168.10.1/32  Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/0
192.168.20.0/24     Static  60   0           RD  10.0.13.3       Serial1/0/1
```

可以观察到，在 R1 的路由表中存在以主机 PC-2 所在网段为目的网段的路由条目，且下一跳路由器为 R3。

测试主机 PC-1 与主机 PC-2 之间的连通性。

```
PC>ping 192.168.20.20
Ping 192.168.20.20: 32 data bytes, Press Ctrl_C to break
From 192.168.20.20: bytes=32 seq=1 ttl=126 time=31 ms
From 192.168.20.20: bytes=32 seq=2 ttl=126 time=32 ms
From 192.168.20.20: bytes=32 seq=3 ttl=126 time=31 ms
From 192.168.20.20: bytes=32 seq=4 ttl=126 time=62 ms
From 192.168.20.20: bytes=32 seq=5 ttl=126 time=47 ms
--- 192.168.20.20 ping statistics ---
```

```
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 31/40/62 ms
```

通信正常，这时可以通过在主机 PC-1 上使用 **tracert** 命令测试所经过的网关。

```
PC>tracert 192.168.20.20
traceroute to 192.168.20.20, 8 hops max
(ICMP), press Ctrl+C to stop
 1  192.168.10.1      16 ms   15 ms   16 ms
 2  10.0.13.3     47 ms   15 ms   31 ms
 3  192.168.20.20     47 ms   47 ms   16 ms
```

通过观察发现数据包是经过 R1 和 R3 到达主机 PC-2 的。

同样在主机 PC-2 和 R3 上进行查看，首先在 R3 上查看路由表。

```
[R3]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 11          Routes : 11
Destination/Mask     Proto    Pre    Cost        Flags NextHop         Interface
      10.0.13.0/24     Direct   0     0           D    10.0.13.3        Serial1/0/1
      10.0.13.1/32     Direct   0     0           D    10.0.13.1        Serial1/0/1
      10.0.13.3/32     Direct   0     0           D    127.0.0.1        Serial1/0/1
      10.0.23.0/24     Direct   0     0           D    10.0.23.3        Serial1/0/0
      10.0.23.2/32     Direct   0     0           D    10.0.23.2        Serial1/0/0
      10.0.23.3/32     Direct   0     0           D    127.0.0.1        Serial1/0/0
      127.0.0.0/8      Direct   0     0           D    127.0.0.1        InLoopBack0
      127.0.0.1/32     Direct   0     0           D    127.0.0.1        InLoopBack0
      192.168.10.0/24  Static   60    0           RD   10.0.13.1        Serial0/0/1
      192.168.20.0/24  Direct   0     0           D    192.168.20.1     GigabitEthernet0/0/0
      192.168.20.1/32  Direct   0     0           D    127.0.0.1        GigabitEthernet0/0/0
```

在 R3 的路由表中存在以主机 PC-1 所在网段为目的网段的路由条目，且下一跳路由器为 R1。

在主机 PC-2 上测试与主机 PC-1 的连通性。

```
PC>ping 192.168.10.10
Ping 192.168.10.10: 32 data bytes, Press Ctrl_C to break
From 192.168.10.10: bytes=32 seq=1 ttl=126 time=32 ms
From 192.168.10.10: bytes=32 seq=2 ttl=126 time=31 ms
From 192.168.10.10: bytes=32 seq=3 ttl=126 time=46 ms
From 192.168.10.10: bytes=32 seq=4 ttl=126 time=31 ms
From 192.168.10.10: bytes=32 seq=5 ttl=126 time=16 ms
--- 192.168.10.10 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 16/31/46 ms
```

可以观察到通信正常。在主机 PC-2 上测试访问主机 PC-1 所经过的网关。

```
PC>tracert 192.168.10.10
traceroute to 192.168.10.10, 8 hops max
(ICMP), press Ctrl+C to stop
 1  192.168.20.1      16 ms   <1 ms   15 ms
 2  10.0.13.1          47 ms   16 ms   31 ms
 3  192.168.10.10     31 ms   31 ms   16 ms
```

可以验证数据包是经过 R3 和 R1 到达主机 PC-1 的。

在总部路由器 R2 上测试与分部的连通性。

```
[R2]ping 192.168.10.10
  PING 192.168.10.10: 56   data bytes, press CTRL_C to break
    Reply from 192.168.10.10: bytes=56 Sequence=1 ttl=127 time=50 ms
    Reply from 192.168.10.10: bytes=56 Sequence=2 ttl=127 time=40 ms
    Reply from 192.168.10.10: bytes=56 Sequence=3 ttl=127 time=30 ms
    Reply from 192.168.10.10: bytes=56 Sequence=4 ttl=127 time=50 ms
    Reply from 192.168.10.10: bytes=56 Sequence=5 ttl=127 time=60 ms
  --- 192.168.10.10 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 30/46/60 ms

[R2]ping 192.168.20.20
  PING 192.168.20.20: 56   data bytes, press CTRL_C to break
    Reply from 192.168.20.20: bytes=56 Sequence=1 ttl=127 time=40 ms
    Reply from 192.168.20.20: bytes=56 Sequence=2 ttl=127 time=40 ms
    Reply from 192.168.20.20: bytes=56 Sequence=3 ttl=127 time=50 ms
    Reply from 192.168.20.20: bytes=56 Sequence=4 ttl=127 time=40 ms
Reply from 192.168.20.20: bytes=56 Sequence=5 ttl=127 time=10 ms
  --- 192.168.20.20 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 10/36/50 ms
```

通过测试，总部路由器 R2 能够正常访问两个分部主机 PC-1 和主机 PC-2 的网络。

3. 配置浮动静态路由实现路由备份

通过上一步骤的配置，现在网络搭建已经初步完成。现需要实现当两分部间通信时，直连链路为主用链路，通过总部的链路为备用链路，即当主用链路发生故障时，可以使用备用链路保障两分部网络间的通信。这里使用浮动静态路由实现网络冗余。

在 R1 上配置静态路由，目的网段为主机 PC-2 所在网段，掩码为 24 位，下一跳为 R2，将路由优先级设置为 100（默认是 60）。

```
[R1]ip route-static 192.168.20.0 24 10.0.12.2 preference 100
```

配置完成后，查看路由器 R1 的路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 12        Routes : 12
Destination/Mask    Proto   Pre  Cost   Flags NextHop        Interface
      10.0.12.0/24  Direct  0    0        D   10.0.12.1      Serial1/0/0
      10.0.12.1/32  Direct  0    0        D   127.0.0.1      Serial1/0/0
      10.0.12.2/32  Direct  0    0        D   10.0.12.2      Serial1/0/0
      10.0.13.0/24  Direct  0    0        D   10.0.13.1      Serial1/0/1
      10.0.13.1/32  Direct  0    0        D   127.0.0.1      Serial1/0/1
```

```
      10.0.13.3/32  Direct  0    0      D   10.0.13.3      Serial1/0/1
      10.0.23.0/24  Static  60   0     RD   10.0.12.2      Serial1/0/0
      127.0.0.0/8   Direct  0    0      D   127.0.0.1      InLoopBack0
      127.0.0.1/32  Direct  0    0      D   127.0.0.1      InLoopBack0
   192.168.10.0/24  Direct  0    0      D   192.168.10.1   GigabitEthernet0/0/0
   192.168.10.1/32  Direct  0    0      D   127.0.0.1      GigabitEthernet0/0/0
   192.168.20.0/24  Static  60   0     RD   10.0.13.3      Serial0/0/1
```

发现路由表此时没有发生任何变化，使用 **display ip routing-table protocol static** 命令仅查看静态路由的路由信息。

```
[R1]display ip routing-table protocol static
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Public routing table : Static
         Destinations : 1         Routes : 2        Configured Routes : 2
Static routing table status :<Active>
         Destinations : 1         Routes : 1
Destination/Mask     Proto   Pre  Cost      Flags NextHop         Interface
192.168.20.0/24  Static  60   0          RD    10.0.13.3        Serial1/0/1
Static routing table status : <Inactive>
         Destinations : 1         Routes : 1
Destination/Mask     Proto   Pre  Cost      Flags NextHop         Interface
 192.168.20.0/24  Static  100  0          R     10.0.12.2        Serial1/0/0
```

可以观察到目的地址为 PC-2 所在网段的两条优先级为 100 和 60 的静态路由条目都已经存在。

现在 R1 上去往相同的目的网段存在有两条不同路由条目，首先会比较它们的优先级，优先级高的，即对应的优先级数值较小的路由条目将被选为主用路由。通过比较，优先级数值为 60 的条目优先级更高，将被 R1 使用，放入路由表中，状态为 Active；而另一条路由状态则为 Inactive，作为备份，不会被放入路由表。只有当 Active 的路由条目失效时，优先级为 100 的路由条目才会被放入路由表。

在 R3 上做和 R1 同样的对称配置。

```
[R3]ip route-static 192.168.10.0 24 10.0.23.2 preference 100
```

接下来，将路由器 R1 的 S 1/0/1 接口关闭，验证使用备份链路。

```
[R1]interface serial 1/0/1
[R1-Serial1/0/1]shutdown
```

配置完成后，查看路由器 R1 的路由表，并使用 **display ip routing-table protocol static** 命令查看。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 9         Routes : 9
Destination/Mask    Proto   Pre  Cost   Flags NextHop         Interface
      10.0.12.0/24  Direct  0    0      D   10.0.12.1      Serial1/0/0
      10.0.12.1/32  Direct  0    0      D   127.0.0.1      Serial1/0/0
      10.0.12.2/32  Direct  0    0      D   10.0.12.2      Serial1/0/0
      10.0.23.0/24  Static  60   0     RD   10.0.12.2      Serial1/0/0
```

```
        127.0.0.0/8   Direct  0    0     D   127.0.0.1        InLoopBack0
       127.0.0.1/32   Direct  0    0     D   127.0.0.1        InLoopBack0
    192.168.10.0/24   Direct  0    0     D   192.168.10.1     GigabitEthernet0/0/0
    192.168.10.1/32   Direct  0    0     D   127.0.0.1        GigabitEthernet0/0/0
    192.168.20.0/24   Static  100  0     RD  10.0.12.2        Serial1/0/0
```

可以观察到，此时优先级为 100 的路由条目已经添加到路由表中。

```
[R1]display ip routing-table protocol static
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Public routing table : Static
         Destinations : 1         Routes : 2          Configured Routes : 2
Static routing table status : <Active>
         Destinations : 1         Routes : 1
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
192.168.20.0/24   Static   100  0          RD   10.0.12.2        Serial1/0/0
Static routing table status : <Inactive>
         Destinations : 1         Routes : 1
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
192.168.20.0/24   Static   60   0                10.0.13.3       Unknown
```

可以观察到，现在优先级为 100 的条目为 Active 状态，优先级为 60 的条目为 Inactive 状态。

测试主机 PC-1 与 PC-2 间的通信。

```
PC>ping 192.168.20.20
Ping 192.168.20.20: 32 data bytes, Press Ctrl_C to break
From 192.168.20.20: bytes=32 seq=1 ttl=125 time=63 ms
From 192.168.20.20: bytes=32 seq=2 ttl=125 time=31 ms
From 192.168.20.20: bytes=32 seq=3 ttl=125 time=47 ms
From 192.168.20.20: bytes=32 seq=4 ttl=125 time=46 ms
From 192.168.20.20: bytes=32 seq=5 ttl=125 time=62 ms
--- 192.168.20.20 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 31/49/63 ms
```

通信正常，再使用 **tracert** 命令查看此时 PC-1 与 PC-2 通信时所经过的网关。

```
PC>tracert 192.168.20.20
traceroute to 192.168.20.20, 8 hops max
(ICMP), press Ctrl+C to stop
 1  192.168.10.1    <1 ms   <1 ms   31 ms
 2  10.0.12.2    32 ms   31 ms   15 ms
 3  10.0.23.3    62 ms   47 ms   47 ms
 4  192.168.20.20     62 ms   47 ms   31 ms
```

再次验证了此时两分部之间通信时已经使用了备用链路。

4. 通过负载均衡实现网络优化

公司网络管理员发现分部之间业务往来越来越多，网络流量剧增，主用链路压力非常大，而总部与两分部间的网络流量相对较少，即备用链路上的带宽多处在闲置状态。此时可以通过配置实现负载均衡，即同时利用主备两条链路来支撑两分部

间的通信。

恢复 R1 上的 S 1/0/1 接口，并配置目的网段为主机 PC-2 所在网段，掩码为 24 位，下一跳为 R2，优先级不变。

```
[R1]interface serial 1/0/1
[R1-Serial1/0/1]undo shutdown
[R1-Serial1/0/1]ip route-static 192.168.20.0 24 10.0.12.2
```

使用 **display ip routing-table** 命令查看 R1 上的路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 12        Routes : 13
Destination/Mask    Proto   Pre  Cost    Flags NextHop        Interface
      10.0.12.0/24  Direct  0    0        D    10.0.12.1      Serial1/0/0
      10.0.12.1/32  Direct  0    0        D    127.0.0.1      Serial1/0/0
      10.0.12.2/32  Direct  0    0        D    10.0.12.2      Serial1/0/0
      10.0.13.0/24  Direct  0    0        D    10.0.13.1      Serial1/0/1
      10.0.13.1/32  Direct  0    0        D    127.0.0.1      Serial1/0/1
      10.0.13.3/32  Direct  0    0        D    10.0.13.3      Serial1/0/1
      10.0.23.0/24  Static  60   0        RD   10.0.12.2      Serial1/0/0
      127.0.0.0/8   Direct  0    0        D    127.0.0.1      InLoopBack0
      127.0.0.1/32  Direct  0    0        D    127.0.0.1      InLoopBack0
   192.168.10.0/24  Direct  0    0        D    192.168.10.1   GigabitEthernet0/0/0
   192.168.10.1/32  Direct  0    0        D    127.0.0.1      GigabitEthernet0/0/0
192.168.20.0/24     Static  60   0        RD   10.0.13.3      Serial1/0/1
                    Static  60   0        RD   10.0.12.2      Serial1/0/0
```

配置完成后，可以观察到现在去往 192.168.20.0 网段拥有两条下一跳不同的路由条目，即实现了负载均衡。

测试主机 PC-1 与 PC-2 间的通信。

```
PC>ping 192.168.20.20
Ping 192.168.20.20: 32 data bytes, Press Ctrl_C to break
From 192.168.20.20: bytes=32 seq=1 ttl=126 time=31 ms
From 192.168.20.20: bytes=32 seq=2 ttl=126 time=31 ms
From 192.168.20.20: bytes=32 seq=3 ttl=126 time=32 ms
From 192.168.20.20: bytes=32 seq=4 ttl=126 time=47 ms
From 192.168.20.20: bytes=32 seq=5 ttl=126 time=31 ms
--- 192.168.20.20 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 31/34/47 ms
```

可以观察到，通信正常。

在 R3 上做和 R1 同样的对称配置。

```
[R3]ip route-static 192.168.10.0 24 10.0.23.2
```

配置完成后，能够在 R3 的路由表中观察到与 R1 路由表相同的情况。

通过配置针对相同目的地址但优先级值不同的静态路由，可以在路由器上实现路径备份的功能。而通过配置针对相同目的地址且优先级值相同的静态路由，不仅互为备份

还能实现负载均衡。

思考

在本实验的步骤 3 和步骤 4 中，如果不在 R3 上做和 R1 同样的对称配置，会产生什么样的现象？为什么？

完成负载均衡的配置之后，可以在 R1 上的 S 1/0/0 和 S 1/0/1 两个接口上启用抓包工具，且在主机 PC-1 上 ping 主机 PC-2，观察 R1 的两个接口上的现象，解释为什么会产生这样的现象。

第7章 RIP

7.1 RIP路由协议基本配置

7.2 配置RIPv2的认证

7.3 RIP路由协议的汇总

7.4 配置RIP的版本兼容、定时器及协议优先级

7.5 配置RIP抑制接口及单播更新

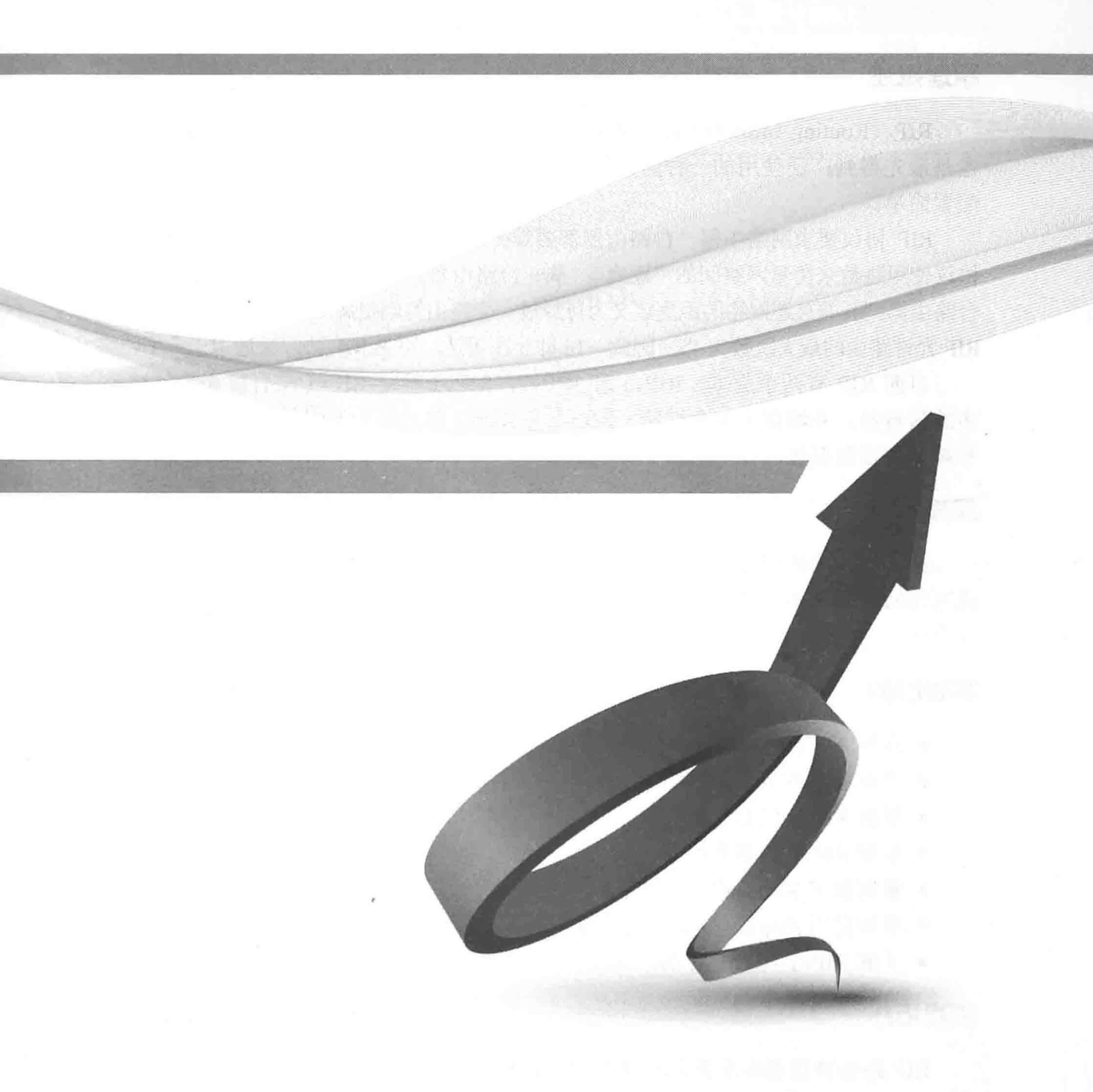
7.6 RIP与不连续子网

7.7 RIP的水平分割及触发更新

7.8 配置RIP路由附加度量值

7.9 RIP的故障处理

7.10 RIP的路由引入



7.1 RIP 路由协议基本配置

原理概述

RIP（Routing Information Protocol，路由协议）作为最早的距离矢量 IP 路由协议，也是最先得到广泛使用的一种路由协议，采用了 Bellman-Ford 算法，其最大的特点就是配置简单。

RIP 协议要求网络中每一台路由器都要维护从自身到每一个目的网络的路由信息。RIP 协议使用跳数来衡量网络间的“距离”：从一台路由器到其直连网络的跳数定义为 1，从一台路由器到其非直连网络的距离定义为每经过一个路由器则距离加 1。“距离”也称为“跳数”。RIP 允许路由的最大跳数为 15，因此，16 即为不可达。可见 RIP 协议只适用于小型网络。

目前 RIP 有两个版本，RIPv1 和 RIPv2，RIPv2 针对 RIPv1 进行扩充，能够携带更多的信息量，并增强了安全性能。RIPv1 和 RIPv2 都是基于 UDP 的协议，使用 UDP520 号端口收发数据包。

实验内容

某小型公司组网拓扑很简单，只拥有两台路由器，因此可以采用 RIP 路由协议来完成网络的部署。本实验通过模拟简单的企业网络场景来描述 RIP 路由协议的基本配置，并介绍一些基本的查看 RIP 信息的命令使用方法。

实验目的

- 理解 RIP 的应用场景
- 理解 RIP 的基本原理
- 掌握 RIPv1 的基本配置
- 掌握 RIPv2 的基本配置
- 掌握测试 RIP 路由网络的连通性的方法
- 掌握使用 **display** 与 **debug** 命令测试 RIP
- 了解 RIPv1 与 RIPv2 的区别

实验拓扑

RIP 路由协议基本配置的拓扑如图 7-1 所示。

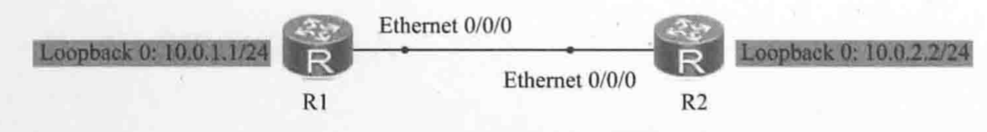


图 7-1 RIP 路由协议基本配置拓扑

实验编址

实验编址见表 7-1。

表 7-1 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR1220）	E 0/0/0	10.0.12.1	255.255.255.0	N/A
	Loopback 0	10.0.1.1	255.255.255.0	N/A
R2（AR1220）	E 0/0/0	10.0.12.2	255.255.255.0	N/A
	Loopback 0	10.0.2.2	255.255.255.0	N/A

实验步骤

1. 基本配置

根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测直连链路的连通性。

```
[R1]ping -c 1 10.0.12.2
  PING 10.0.12.2: 56   data bytes, press CTRL_C to break
 Reply from 10.0.12.2: bytes=56 Sequence=1 ttl=255 time=50 ms
  --- 10.0.12.2 ping statistics ---
    1 packet(s) transmitted
    1 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 50/50/50 ms
```

2. 使用 RIPv1 搭建网络

在公司两台路由器 R1 和 R2 上配置 RIP v1。使用 **rip** 命令创建并开启协议进程，默认情况下进程号是 1。使用 **network** 命令对指定网段接口使能 RIP 功能，注意必须是自然网段的地址。

```
[R1]rip
[R1-rip-1]network 10.0.0.0

[R2]rip
[R2-rip-1]network 10.0.0.0
```

配置完成后，使用 **display ip routing-table** 命令查看 R1、R2 的路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
Destinations : 7        Routes : 7
Destination/Mask   Proto   Pre   Cost    Flags  NextHop       Interface
10.0.1.0/24        Direct  0     0       D      10.0.1.1      LoopBack0
10.0.1.1/32        Direct  0     0       D      127.0.0.1     LoopBack0
10.0.2.0/24        RIP     100   1       D      10.0.12.2     Ethernet0/0/0
10.0.12.0/24       Direct  0     0       D      10.0.12.1     Ethernet0/0/0
10.0.12.1/32       Direct  0     0       D      127.0.0.1     Ethernet0/0/0
127.0.0.0/8        Direct  0     0       D      127.0.0.1     InLoopBack0
127.0.0.1/32       Direct  0     0       D      127.0.0.1     InLoopBack0

[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
Destinations : 7        Routes : 7
```

```
Destination/Mask    Proto   Pre   Cost    Flags   NextHop     Interface
10.0.1.0/24         RIP     100   1       D       10.0.12.1   Ethernet0/0/0
10.0.2.0/24         Direct  0     0       D       10.0.2.2    LoopBack0
10.0.2.2/32         Direct  0     0       D       127.0.0.1   LoopBack0
10.0.12.0/24        Direct  0     0       D       10.0.12.2   Ethernet0/0/0
10.0.12.2/32        Direct  0     0       D       127.0.0.1   Ethernet0/0/0
127.0.0.0/8         Direct  0     0       D       127.0.0.1   InLoopBack0
127.0.0.1/32        Direct  0     0       D       127.0.0.1   InLoopBack0
```

可以观察到，两台路由器已经通过 RIP 协议学习到了对方环回接口所在网段的路由条目。

测试 R1 与 R2 环回接口间的连通性。

```
[R1]ping 10.0.2.2
  PING 10.0.2.2: 56    data bytes, press CTRL_C to break
    Reply from 10.0.2.2: bytes=56 Sequence=1 ttl=255 time=1 ms
    Reply from 10.0.2.2: bytes=56 Sequence=2 ttl=255 time=10 ms
    Reply from 10.0.2.2: bytes=56 Sequence=3 ttl=255 time=10 ms
    Reply from 10.0.2.2: bytes=56 Sequence=4 ttl=255 time=20 ms
    Reply from 10.0.2.2: bytes=56 Sequence=5 ttl=255 time=10 ms
  --- 10.0.2.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 1/10/20 ms
```

可以观察到通信正常。

使用 **debuging** 命令查看 RIP 协议定期更新情况，并开启 RIP 调试功能。请注意，**debug** 命令需要在用户视图下才能使用。使用 **terminal debugging** 和 **terminal monitor** 命令开启 debug 信息在屏幕上显示的功能，才能在电脑屏幕上看到路由器之间 RIP 协议交互的信息。

```
<R1>debugging rip 1
<R1>terminal debugging
Info: Current terminal debugging is on.
<R1>terminal monitor
Info: Current terminal monitor is on.
<R1>
Dec  7 2012 11:20:22.530.1-08:00 R1 RIP/7/DBG: 6: 12176: RIP 1: Sending v1 response on Ethernet0/0/0 from 10.0.12.1
with 1 RTE
Dec  7 2012 11:20:22.530.2-08:00 R1 RIP/7/DBG: 6: 12227: RIP 1: Sending response
 on interface Ethernet0/0/0 from 10.0.12.1 to 255.255.255.255
Dec  7 2012 11:20:22.530.3-08:00 R1 RIP/7/DBG: 6: 12247: Packet: Version 1, Cmd response, Length 24
Dec  7 2012 11:20:22.530.4-08:00 R1 RIP/7/DBG: 6: 12296: Dest 10.0.1.0, Cost 1
Dec  7 2012 11:20:24.510.1-08:00 R1 RIP/7/DBG: 6: 12185: RIP 1: Receiving v1 response on Ethernet0/0/0 from 10.0.12.2
with 1 RTE
Dec  7 2012 11:20:24.510.2-08:00 R1 RIP/7/DBG: 6: 12236: RIP 1: Receive response
 from 10.0.12.2 on Ethernet0/0/0
Dec  7 2012 11:20:24.510.3-08:00 R1 RIP/7/DBG: 6: 12247: Packet: Version 1, Cmd response, Length 24
Dec  7 2012 11:20:24.510.4-08:00 R1 RIP/7/DBG: 6: 12296: Dest 10.0.2.0, Cost 1
```

可以观察到 R1 从连接 R2 的 E 0/0/0 接口周期性发送、接收 v1 的 Response 更新报文，包括目的地、数据包大小以及 cost 值。

可以使用 **undo debuging rip** 或者 **undo debug all** 命令关闭 debug 调试功能。

```
<R1>undo debugging rip 1
```

也可以使用带更多参数的命令查看某类型的调试信息，如 **debuging rip 1 event** 查看路由器发出和收到的定期更新事件。其他参数可以使用“？”获取帮助。

```
<R1>debugging rip 1 event
Dec  7 2012 11:21:18.690.1-08:00 R1 RIP/7/DBG: 25: 4379: RIP 1: Periodic timer expired for interface Ethernet0/0/0
(10.0.12.1) and its added to periodic update
queue
Dec  7 2012 11:21:18.690.2-08:00 R1 RIP/7/DBG: 25: 4707: RIP 1: Interface Ethern
et0/0/0 (10.0.12.1) is deleted from the periodic update queue
<R1>undo debugging all
Info: All possible debugging has been turned off.
```

提示 开启过多的 debug 功能会耗费大量路由器资源，甚至可能导致宕机。请慎重使用开启批量 debug 功能的命令，如 **debug all**。

3. 使用 RIPv2 搭建网络

基于前面的配置，现在只需在 RIP 子视图模式下配置 **v2** 即可。

```
[R1]rip
[R1-rip-1]version 2

[R2]rip
[R2-rip-1]version 2
```

配置完成后使用 **display ip routing-table** 命令查看各路由器路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7          Routes : 7
Destination/Mask    Proto   Pre    Cost      Flags   NextHop        Interface
10.0.1.0/24         Direct  0      0         D       10.0.1.1       LoopBack0
10.0.1.1/32         Direct  0      0         D       127.0.0.1      LoopBack0
10.0.2.0/24         RIP     100    1         D       10.0.12.2      Ethernet0/0/0
10.0.12.0/24        Direct  0      0         D       10.0.12.1      Ethernet0/0/0
10.0.12.1/32        Direct  0      0         D       127.0.0.1      Ethernet0/0/0
127.0.0.0/8         Direct  0      0         D       127.0.0.1      InLoopBack0
127.0.0.1/32        Direct  0      0         D       127.0.0.1      InLoopBack0

[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7          Routes : 7
Destination/Mask    Proto   Pre    Cost      Flags      NextHop      Interface
10.0.1.0/24         RIP     100    1         D          10.0.12.1    Ethernet0/0/0
10.0.2.0/24         Direct  0      0         D          10.0.2.2     LoopBack0
10.0.2.2/32         Direct  0      0         D          127.0.0.1    LoopBack0
10.0.12.0/24        Direct  0      0         D          10.0.12.2    Ethernet0/0/0
10.0.12.2/32        Direct  0      0         D          127.0.0.1    Ethernet0/0/0
127.0.0.0/8         Direct  0      0         D          127.0.0.1    InLoopBack0
127.0.0.1/32        Direct  0      0         D          127.0.0.1    InLoopBack0
```

可以观察到，两台路由器已经通过 RIP 协议学习到了对方环回接口所在网段的路由条目。

配置完成后，使用 **ping** 命令检测 R1 与 R2 之间直连链路的 IP 连通性。

```
[R1]ping 10.0.2.2
```

```
PING 10.0.2.2: 56   data bytes, press CTRL_C to break
  Reply from 10.0.2.2: bytes=56 Sequence=1 ttl=255 time=50 ms
  Reply from 10.0.2.2: bytes=56 Sequence=2 ttl=255 time=10 ms
  Reply from 10.0.2.2: bytes=56 Sequence=3 ttl=255 time=40 ms
  Reply from 10.0.2.2: bytes=56 Sequence=4 ttl=255 time=30 ms
  Reply from 10.0.2.2: bytes=56 Sequence=5 ttl=255 time=10 ms
--- 10.0.2.2 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 10/28/50 ms
```

可以观察到通信正常。

使用 **debuging** 命令查看 RIPv2 协议定期更新情况。

```
<R1>debugging rip 1
Dec   7 2012 11:37:47.620.3-08:00 R1 RIP/7/DBG: 6: 12247: Packet: Version 2, Cmd
response, Length 24
Dec   7 2012 11:37:47.620.4-08:00 R1 RIP/7/DBG: 6: 12315: Dest 10.0.2.0/24, Nexth
op 0.0.0.0, Cost 1, Tag 0
Dec   7 2012 11:37:48.470.1-08:00 R1 RIP/7/DBG: 6: 12176: RIP 1: Sending v2 respo
nse on Ethernet0/0/0 from 10.0.12.1 with 2 RTEs
Dec   7 2012 11:37:48.470.2-08:00 R1 RIP/7/DBG: 6: 12227: RIP 1: Sending response
 on interface Ethernet0/0/0 from 10.0.12.1 to 224.0.0.9
<R1>undo debugging rip 1
```

与 RIPv1 中使用 **debuging** 命令所查看的信息进行对比，可以明显区分出 RIPv1 和 RIPv2 的不同：

- RIPv2 的路由信息中携带了子网掩码；
- RIPv2 的路由信息中携带了下一跳地址，标识一个比通告路由器的地址更好的下一跳地址。换句话说，它指出的地址，其度量值（跳数）比在同一个子网上的通告路由器更靠近目的地。如果这个字段设置为全 0（0.0.0.0），说明通告路由器的地址是最优的下一跳地址；
- RIPv2 默认采用组播方式发送报文，地址为 224.0.0.9。

7.2 配置 RIPv2 的认证

原理概述

配置协议的认证可以降低设备接受非法路由选择更新消息的可能性，也可称为“验证”。非法的更新消息可能来自试图破坏网络的攻击者，或试图通过欺骗路由器发送数据到错误的目的地址的方法来捕获数据包。RIPv2 协议能够通过更新消息所包含的口令来验证某个路由选择消息源的合法性，有简单和 MD5 密文两种验证方式。

简单验证是指在认证的消息当中所携带的认证口令是以明文传输的，可以通过抓包软件抓取到数据包中的密码。

MD5 密文验证是一种单向消息摘要（message digest）算法或安全散列函数（secure

hash function），由 RSA Date Security,Inc 提出。有时 MD5 也被作为一个加密校验和（cryptographic checksum）。MD5 算法是通过一个随意长度的明文消息（例如，一个 RIPv2 的更新消息）和口令计算出一个 128 位的 hash 值。hash 值类似“指纹”，这个“指纹”随同消息一起传送，拥有相同口令的接收者会计算它自己的 hash 值，如果消息的内容没有被更改，接收者的 hash 值应该和消息发送者的 hash 值相匹配。

实验目的

- 理解配置 RIPv2 认证的场景和意义
- 掌握配置 RIPv2 简单验证的方法
- 掌握测试 RIPv2 简单验证的配置结果的方法
- 掌握配置 RIPv2 MD5 密文验证的方法
- 掌握测试 RIPv2 MD5 密文验证配置结果的方法

实验内容

本实验模拟企业网络场景。某公司有两台路由器 R1 与 R2，各自连接着一台主机，并且 R1 和 R2 之间配置 RIPv2 协议学习路由条目。R3 模拟作为网络中的攻击者，窃取 R1 与 R2 间的路由信息，并发布了一些虚假路由，使 R1 和 R2 的相关路由的选路指向了 R3，形成了路由欺骗。为了避免遭受攻击，提高网络安全性，网络管理员将配置 RIPv2 认证。

实验拓扑

配置 RIPv2 的认证拓扑如图 7-2 所示。

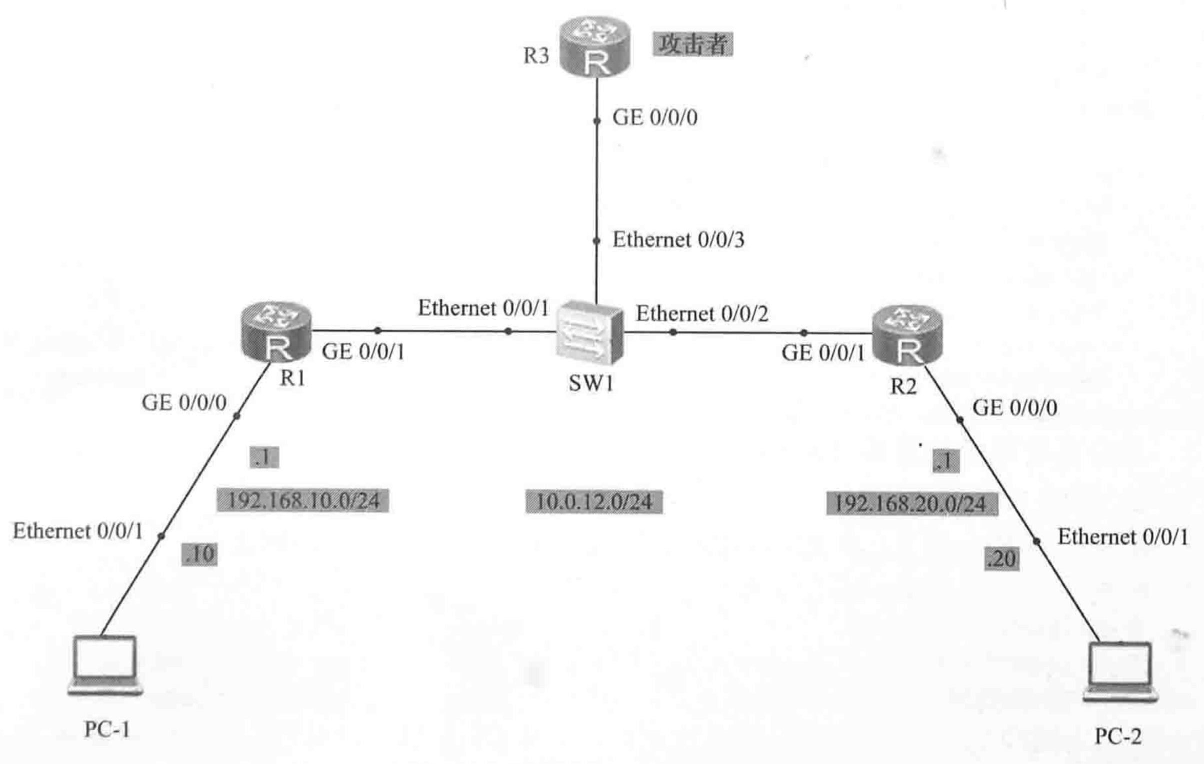


图 7-2 配置 RIPv2 的认证拓扑

实验编址

实验编址见表 7-2。

表 7-2 实验编址

设备	接口	IP 地址	子网掩码	默认网关
PC-1	Ethernet 0/0/1	192.168.10.10	255.255.255.0	192.168.10.1
R1（AR2220）	GE 0/0/0	192.168.10.1	255.255.255.0	N/A
	GE 0/0/1	10.0.12.1	255.255.255.0	N/A
R2（AR2220）	GE 0/0/1	10.0.12.2	255.255.255.0	N/A
	GE 0/0/0	192.168.20.1	255.255.255.0	N/A
PC-2	Ethernet 0/0/1	192.168.20.20	255.255.255.0	192.168.20.1
R3（AR2220）	GE 0/0/0	10.0.12.3	255.255.255.0	N/A
	Loopback 0	192.168.10.1	255.255.255.0	N/A
	Loopback 1	192.168.20.1	255.255.255.0	N/A

实验步骤

1. 基本配置

根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测各直连链路的连通性。其中，R3 上的两个环回接口先不配置 IP 地址。

```
[R1]ping -c 1 10.0.12.2
  PING 10.0.12.2: 56   data bytes, press CTRL_C to break
    Reply from 10.0.12.2: bytes=56 Sequence=1 ttl=255 time=50 ms
  --- 10.0.12.2 ping statistics ---
    1 packet(s) transmitted
    1 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 50/50/50 ms

[R1]ping -c 1 192.168.10.10
  PING 192.168.10.10: 56   data bytes, press CTRL_C to break
    Reply from 192.168.10.10: bytes=56 Sequence=1 ttl=128 time=30 ms
  --- 192.168.10.10 ping statistics ---
    1 packet(s) transmitted
    1 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 30/30/30 ms
```

其余直连网段的连通性测试省略。

2. 搭建 RIP 网络

配置公司路由器 R1 和 R2 的 RIPv2 协议，并添加需要通告的网段。

```
[R1]rip 1
[R1-rip-1]version 2
[R1-rip-1]network 192.168.10.0
[R1-rip-1]network 10.0.0.0

[R2]rip 1
[R2-rip-1]version 2
```

```
[R2-rip-1]network 192.168.20.0
[R2-rip-1]network 10.0.0.0
```

配置完成后，检查 R1 与 R2 的路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
Destinations : 8        Routes : 8
Destination/Mask    Proto   Pre  Cost  Flags  NextHop        Interface
    10.0.12.0/24    Direct  0    0     D      10.0.12.1      GigabitEthernet0/0/1
    127.0.0.0/8     Direct  0    0     D      127.0.0.1      InLoopBack0
    127.0.0.1/32    Direct  0    0     D      127.0.0.1      InLoopBack0
    192.168.10.0/24 Direct  0    0     D      192.168.10.1   GigabitEthernet0/0/0
    192.168.10.1/32 Direct  0    0     D      127.0.0.1      GigabitEthernet0/0/0
    192.168.20.0/24 RIP     100  1     D      10.0.12.2      GigabitEthernet0/0/1

[R2-rip-1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
Destinations : 8        Routes : 8
Destination/Mask    Proto   Pre  Cost  Flags  NextHop        Interface
    10.0.12.0/24    Direct  0    0     D      10.0.12.2      GigabitEthernet0/0/1
    127.0.0.0/8     Direct  0    0     D      127.0.0.1      InLoopBack0
    127.0.0.1/32    Direct  0    0     D      127.0.0.1      InLoopBack0
    192.168.10.0/24 RIP     100  1     D      10.0.12.1      GigabitEthernet0/0/1
    192.168.20.0/24 Direct  0    0     D      192.168.20.1   GigabitEthernet0/0/0
   192.168.20.1/32  Direct  0    0     D      127.0.0.1      GigabitEthernet0/0/0
```

可以观察到，此时双方已经正常地获得了 RIP 路由条目。

3. 模拟网络攻击

配置路由器 R3 作为攻击者，接入公司网络。在基本配置中已经将接口 GE 0/0/0 地址配置为 10.0.12.3，与该公司路由器在同一网段，并配置 RIPv2 协议，通告该网段，配置完成后查看 R3 的路由表。

```
[R3]rip 1
[R3-rip-1]version 2
[R3-rip-1]network 10.0.0.0

[R3-rip-1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
Destinations : 5        Routes : 5
Destination/Mask    Proto   Pre  Cost  Flags  NextHop        Interface
    10.0.12.0/24    Direct  0    0     D      10.0.12.2      GigabitEthernet0/0/0
    127.0.0.0/8     Direct  0    0     D      127.0.0.1      InLoopBack0
    127.0.0.1/32    Direct  0    0     D      127.0.0.1      InLoopBack0
    192.168.10.0/24 RIP     100  1     D      10.0.12.1      GigabitEthernet0/0/0
    192.168.20.0/24 RIP     100  1     D      10.0.12.2      GigabitEthernet0/0/0
```

观察发现 R3 已经非法获取了 R1 和 R2 上用户终端所在的两个网段的路由信息。此时 R3 就可以向两个网段发送大量的 ping 包，导致网络链路拥塞，形成攻击。

下面是 R3 模拟攻击演示，发送 10 万个 ping 包给 PC-1，导致攻击发生。可以按<Ctrl+C>组合键来终止该操作。

```
[R3]ping   -c 100000 192.168.10.10
  PING 192.168.20.1: 56 data bytes, press CTRL_C to break
    Reply from 192.168.10.10: bytes=56 Sequence=1 ttl=255 time=480 ms
    Reply from 192.168.10.10: bytes=56 Sequence=2 ttl=255 time=170 ms
    Reply from 192.168.10.10: bytes=56 Sequence=3 ttl=255 time=130 ms
    Reply from 192.168.10.10: bytes=56 Sequence=4 ttl=255 time=30 ms
    Reply from 192.168.10.10: bytes=56 Sequence=5 ttl=255 time=70 ms
    Reply from 192.168.10.10: bytes=56 Sequence=6 ttl=255 time=60 ms
```

完成上述模拟后，在 R3 上分别配置两个用于欺骗的环回接口，地址分别为 192.168.10.1 和 192.168.20.1，即与公司网络中两个用户的地址相同，并且在 RIP 协议中通告这两个欺骗的网段。

```
[R3]interface loopback0
[R3-LoopBack0]ip address 192.168.10.1 255.255.255.0
[R3-LoopBack0]interface loopback1
[R3-LoopBack1]ip address 192.168.20.1 255.255.255.0
[R3-LoopBack1]rip 1
[R3-rip-1]version 2
[R3-rip-1]network 192.168.10.0
[R3-rip-1]network 192.168.20.0
```

配置完成后，查看 R1 与 R2 的路由表。

```
[R1]display ip routing-table
Destination/Mask Proto Pre Cost Flags NextHop       Interface
......
  192.168.20.0/24  RIP    100    1     D      10.0.12.2       GigabitEthernet0/0/1
                   RIP    100    1     D      10.0.12.3       GigabitEthernet0/0/1
[R2]display ip routing-table
Destination/Mask Proto Pre Cost Flags NextHop       Interface
......
  192.168.10.0/24  RIP    100    1     D      10.0.12.1       GigabitEthernet0/0/1
                   RIP    100    1     D      10.0.12.3       GigabitEthernet0/0/1
```

可以观察到 R3 发过来的路由更新。因为 R2 和 R3 发送 RIP 更新的 cost 都是 1 跳，所以在 R1 的路由表中，目的为 192.168.20.0 网段形成了两条等价负载均衡的路径，下一跳分别是 R2 和 R3。这样会导致去往 192.168.20.0 网段的数据包有部分转发给了欺骗路由器 R3，R2 的路由表变化和 R1 同理。

4. 配置 RIPv2 简单验证

为了提升网络安全性，避免发生上述攻击和路由欺骗，网络管理员可在 R1 和 R2 上配置 RIP 的简单验证实现对网络的保护。

在路由器 R1 和 R2 的 GE 0/0/1 接口配置认证，使用简单验证方式，密码为 huawei。注意，两端的密码必须保持一致，否则会导致认证失败，从而使得 RIP 协议无法正常运行。

```
[R1]interface GigabitEthernet 0/0/1
[R1-GigabitEthernet0/0/1]rip authentication-mode simple huawei

[R2]interface GigabitEthernet0/0/1
[R2- GigabitEthernet0/0/1]rip authentication-mode simple huawei
```

配置完成后，等待一段时间，再次查看 R1 和 R2 的路由表。

```
[R1]display ip routing-table
Destination/Mask    Proto   Pre   Cost   Flags   NextHop        Interface
   10.0.12.0/24     Direct  0     0      D       10.0.12.1      GigabitEthernet0/0/1
   127.0.0.0/8      Direct  0     0      D       127.0.0.1      InLoopBack0
   127.0.0.1/32     Direct  0     0      D       127.0.0.1      InLoopBack0
   192.168.10.0/24  Direct  0     0      D       192.168.10.1   GigabitEthernet0/0/0
   192.168.10.1/32  Direct  0     0      D       127.0.0.1      GigabitEthernet0/0/0
192.168.20.0/24     RIP     100   1      D       10.0.12.2      GigabitEthernet0/0/1

[R2]display ip routing-table
Destination/Mask    Proto   Pre   Cost   Flags   NextHop        Interface
   10.0.12.0/24     Direct  0     0      D       10.0.12.2      GigabitEthernet0/0/1
   127.0.0.0/8      Direct  0     0      D       127.0.0.1      InLoopBack0
   127.0.0.1/32     Direct  0     0      D       127.0.0.1      InLoopBack0
192.168.10.0/24     RIP     100   1      D       10.0.12.1      GigabitEthernet0/0/1
   192.168.20.0/24  Direct  0     0      D       192.168.20.1   GigabitEthernet0/0/0
192.168.20.1/32     Direct  0     0      D       127.0.0.1      GigabitEthernet0/0/0
```

可以观察到现在 R1 与 R2 的路由表恢复正常，R3 发送的欺骗路由在路由表中消失。原因是 R1 和 R2 配置了 RIP 认证，就要求在 RIP 更新报文中包含认证密码，如果密码错误或者不存在，将认为该路由非法并丢弃。

在路由器 R1 的 GE 0/0/1 接口上抓包，如图 7-3 所示。

```
19 34.632000000 10.0.12.1            224.0.0.9           RIPv2     76 Response

Frame 19: 76 bytes on wire (608 bits), 76 bytes captured (608 bits) on interface 0
Point-to-Point Protocol
Internet Protocol Version 4, Src: 10.0.12.1 (10.0.12.1), Dst: 224.0.0.9 (224.0.0.9)
User Datagram Protocol, Src Port: router (520), Dst Port: router (520)
Routing Information Protocol
  Command: Response (2)
  Version: RIPv2 (2)
  Authentication: Simple Password
    Authentication type: Simple Password (2)
    Password: huawei
  IP Address: 192.168.10.0, Metric: 1
```

图 7-3 抓包观察

可以观察到，此时 R1 与 R2 间发送的 RIP 报文中含 authentication 字段，并且密码是明文的 huawei。

5. 配置 RIPv2 MD5 密文验证

在上一步骤中，在 R1 和 R2 上配置了简单验证方式的认证后，成功地抵御了 R3 的路由欺骗和攻击，且主机 PC-1 与 PC-2 可以正常通信。但是通过抓包能够发现，简单验证方式下的认证安全性非常差，攻击者虽然无法直接攻击网络，但是可以通过抓取 RIP 协议数据包获得明文密码，因此建议使用 MD5 密文验证方式进行 RIPv2 的认证。

在 R1 和 R2 的 GE 0/0/1 接口上删除上一步骤中的简单验证配置，选择使用 MD5 密文验证方式配置。配置时可以选择 MD5 密文验证方式的报文格式，usual 参数表示使用通用报文格式；nonstandard 参数表示使用非标准报文格式（IETF 标准），但是必须保证两端的报文格式一致，这里选用通用标准格式。

```
[R1-GigabitEthernet0/0/1]undo rip authentication-mode
[R1-GigabitEthernet0/0/1]rip authentication-mode md5 usual huawei
```

```
[R2-GigabitEthernet0/0/1]undo rip authentication-mode
[R2-GigabitEthernet0/0/1]rip authentication-mode md5 usual huawei
```

配置完成后，查看 R1 和 R2 路由表。

```
[R1]display ip routing-table
Destination/Mask    Proto   Pre  Cost  Flags  NextHop       Interface
    10.0.12.0/24    Direct  0    0     D      10.0.12.1     GigabitEthernet0/0/1
    127.0.0.0/8     Direct  0    0     D      127.0.0.1     InLoopBack0
    127.0.0.1/32    Direct  0    0     D      127.0.0.1     InLoopBack0
    192.168.10.0/24 Direct  0    0     D      192.168.10.1  GigabitEthernet0/0/0
    192.168.10.1/32 Direct  0    0     D      127.0.0.1     GigabitEthernet0/0/0
192.168.20.0/24     RIP     100  1     D      10.0.12.2     GigabitEthernet0/0/1

[R2]display ip routing-table
Destination/Mask    Proto   Pre  Cost  Flags  NextHop       Interface
    10.0.12.0/24    Direct  0    0     D      10.0.12.2     GigabitEthernet0/0/1
    127.0.0.0/8     Direct  0    0     D      127.0.0.1     InLoopBack0
    127.0.0.1/32    Direct  0    0     D      127.0.0.1     InLoopBack0
192.168.10.0/24     RIP     100  1     D      10.0.12.1     GigabitEthernet0/0/1
    192.168.20.0/24 Direct  0    0     D      192.168.20.1  GigabitEthernet0/0/0
192.168.20.1/32     Direct  0    0     D      127.0.0.1     GigabitEthernet0/0/0
```

可以观察到 R1 与 R2 的路由表正常，R3 发送的欺骗路由在路由表中消失，与配置简单验证的效果一样。

继续抓取 R1 的 GE 0/0/1 接口上的报文，如图 7-4 所示。

```
No.   Time      Source     Destination  Protocol  Length  Info
1224  2565.718  10.1.12.1  224.0.0.9    RIPv2     116     Response

⊞ Frame 1224: 116 bytes on wire (928 bits), 116 bytes captured (928 bits) on interface 0
⊞ Point-to-Point Protocol
⊞ Internet Protocol Version 4, Src: 10.1.12.1 (10.1.12.1), Dst: 224.0.0.9 (224.0.0.9)
⊞ User Datagram Protocol, Src Port: router (520), Dst Port: router (520)
⊟ Routing Information Protocol
    Command: Response (2)
    Version: RIPv2 (2)
  ⊟ Authentication: Keyed Message Digest
      Authentication type: Keyed Message Digest (3)
      Digest Offset: 64
      Key ID: 1
      Auth Data Len: 20
      Seq num: 13
      Zero Padding
    ⊟ Authentication Data Trailer
        Authentication Data: db 62 54 d3 5f 3e 19 90 3b b2 4c aa 46 15 71 04
  ⊞ IP Address: 10.1.12.0, Metric: 1
  ⊞ IP Address: 192.168.10.0, Metric: 1
```

图 7-4 抓包观察

抓包发现已经无法看到配置的认证密码，而看到的是一个 128 位的 hash 值，这是一种单向的散列值，难以破解，这样就能够进一步地保证网络的安全性。

思考

在本实验中，R1 和 R2 上配置了认证，R3 没有配置认证，根据分析，R1 和 R2 不会再接收 R3 发送的不包含认证信息的 RIP 更新，那 R3 是否会接收 R1 和 R2 发送过来的带有认证信息的 RIP 更新呢？为什么？

7.3 RIP 路由协议的汇总

原理概述

当网络中路由器的路由条目非常多时，可以通过路由汇总（又称路由汇聚或路由聚合）来减少路由条目数，加快路由收敛时间和增强网络稳定性。路由汇总的原理是，同一个自然网段内的不同子网的路由在向外（其他网段）发送时聚合成一个网段的路由发送。由于汇总后路由器将不会感知被汇总子网有关的变化，从而提高了网络稳定性，减少了不必要的路由器更新。

RIPv1 是有类别路由协议，它的协议报文中没有携带掩码信息，只能识别 A、B、C 类这样的自然网段的路由，因此 RIPv1 无法支持路由聚合，也不支持不连续子网，所有路由会被自动汇总为有类路由。

RIPv2 是一种无分类路由协议，报文中携带掩码信息，支持手动路由汇总和自动路由汇总两种方式。

■ 基于 RIP 进程的有类自动汇总：比如对于 10.1.1.0/24（metric=2）和 10.1.2.0/24（metric=3）这两条路由，聚合成自然网段路由 10.0.0.0/8（metric=2）。自动汇总是按类聚合的，在华为设备上自动汇总是默认关闭的，可手动更改配置使自动汇总生效；

■ 基于接口的手动汇总：用户可以指定聚合路由。比如，对于 10.1.1.0/24（metric=2）和 10.1.2.0/24（metric=3）这两条路由，可以在此接口上配置聚合路由 10.1.0.0/16（metric=2）。

实验目的

- 理解 RIP 路由协议汇总的应用场景
- 理解 RIPv1 和 RIPv2 的自动汇总
- 掌握配置和测试 RIPv2 手动汇总的方法

实验内容

在由 3 台路由器所组成的简单网络中，R3 连接着多个网段，通过 Loopback 口来模拟多个网段，通过实验实现 RIPv1 自动汇总、RIPv2 自动汇总以及 RIPv2 手工汇总。

实验拓扑

RIP 路由协议的汇总拓扑如图 7-5 所示。

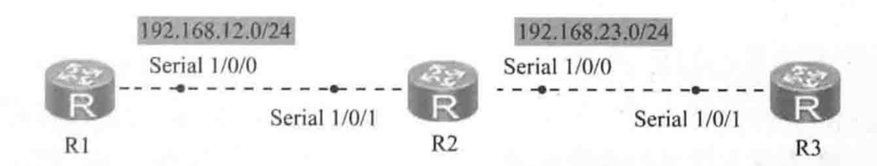


图 7-5 RIP 路由协议的汇总拓扑

实验编址

实验编址见表 7-3。

表 7-3 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR1220）	Serial 1/0/0	192.168.12.1	255.255.255.0	N/A
R2（AR1220）	Serial 1/0/1	192.168.12.2	255.255.255.0	N/A
	Serial 1/0/0	192.168.23.2	255.255.255.0	N/A
R3（AR1220）	Serial 1/0/1	192.168.23.3	255.255.255.0	N/A
	Loopback 0	3.3.0.3	255.255.255.0	N/A
	Loopback 1	3.3.1.3	255.255.255.0	N/A
	Loopback 2	3.3.2.3	255.255.255.0	N/A
	Loopback 3	3.3.3.3	255.255.255.0	N/A

实验步骤

1. 基本配置

根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测各直连链路的连通性。

```
<R1>ping -c 1 192.168.12.2
  PING 192.168.12.2: 56   data bytes, press CTRL_C to break
    Reply from 192.168.12.2: bytes=56 Sequence=1 ttl=255 time=30 ms
  --- 192.168.12.2 ping statistics ---
    1 packet(s) transmitted
    1 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 30/30/30 ms
```

其余直连网段的连通性测试省略。

2. 配置 RIPv1 协议

在路由器 R1、R2、R3 上配置 RIPv1 协议，通告相应网段。

```
[R1]rip 1
[R1-rip-1]network 192.168.12.0

[R2]rip 1
[R2-rip-1]network 192.168.12.0
[R2-rip-1]network 192.168.23.0

[R3]rip 1
[R3-rip-1]network 192.168.23.0
[R3-rip-1]network 3.0.0.0
```

配置完成后，查看 R1 与 R2 的路由表。

```
<R1>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7          Routes : 7
```

```
Destination/Mask    Proto   Pre  Cost   Flags  NextHop         Interface
       3.0.0.0/8    RIP     100  2      D      192.168.12.2    Serial1/0/0
     127.0.0.0/8    Direct  0    0      D      127.0.0.1       InLoopBack0
     127.0.0.1/32   Direct  0    0      D      127.0.0.1       InLoopBack0
  192.168.12.0/24   Direct  0    0      D      192.168.12.1    Serial1/0/0
  192.168.12.1/32   Direct  0    0      D      127.0.0.1       Serial1/0/0
  192.168.12.2/32   Direct  0    0      D      192.168.12.2    Serial1/0/0
  192.168.23.0/24   RIP     100  1      D      192.168.12.2    Serial1/0/0

<R2>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 9        Routes : 9
Destination/Mask    Proto   Pre  Cost   Flags  NextHop         Interface
       3.0.0.0/8    RIP     100  1      D      192.168.23.3    Serial1/0/0
     127.0.0.0/8    Direct  0    0      D      127.0.0.1       InLoopBack0
     127.0.0.1/32   Direct  0    0      D      127.0.0.1       InLoopBack0
  192.168.12.0/24   Direct  0    0      D      192.168.12.2    Serial1/0/1
  192.168.12.1/32   Direct  0    0      D      192.168.12.1    Serial1/0/1
  192.168.12.2/32   Direct  0    0      D      127.0.0.1       Serial1/0/1
  192.168.23.0/24   Direct  0    0      D      192.168.23.2    Serial1/0/0
  192.168.23.2/32   Direct  0    0      D      127.0.0.1       Serial1/0/0
  192.168.23.3/32   Direct  0    0      D      192.168.23.3    Serial1/0/0
```

可以观察到R3发送过来的汇总路由条目3.0.0.0/8，没有任何明细路由条目。

在R3的S 1/0/1接口上抓包，如图7-6所示。

```
No.  Time              Source         Destination      Protocol  Length  TX rate  RSSI  Frequency/channel  Info
59   120.932000000     192.168.23.3   255.255.255.255  RIPv1     76                                        Response
64   124.162000000     192.168.23.2   255.255.255.255  RIPv1     56                                        Response
73   151.134000000     192.168.23.3   255.255.255.255  RIPv1     76                                        Response
78   156.282000000     192.168.23.2   255.255.255.255  RIPv1     56                                        Response

⊞ Frame 59: 76 bytes on wire (608 bits), 76 bytes captured (608 bits) on interface 0
⊞ Point-to-Point Protocol
⊞ Internet Protocol Version 4, Src: 192.168.23.3 (192.168.23.3), Dst: 255.255.255.255 (255.255.255.2
⊞ User Datagram Protocol, Src Port: router (520), Dst Port: router (520)
⊟ Routing Information Protocol
    Command: Response (2)
    Version: RIPv1 (1)
  ⊟ IP Address: 3.0.0.0, Metric: 1
      Address Family: IP (2)
      IP Address: 3.0.0.0 (3.0.0.0)
      Metric: 1
  ⊟ IP Address: 192.168.23.0, Metric: 16
      Address Family: IP (2)
      IP Address: 192.168.23.0 (192.168.23.0)
      Metric: 16
```

图7-6 抓包观察

可以观察到，RIPv1的协议报文中没有携带掩码信息，只有相应的网络号以及Metric值，即RIPv1只发布汇总后的有类路由。RIPv1默认开启自动汇总，且无法关闭，也不支持手动汇总。可以使用**display default-parameter rip**命令查看RIP默认配置信息。

```
<R3>display default-parameter rip
------------------------------------------------------------------------------
 Protocol Level Default Configurations
```

```
-----------------------------------------------------------------------
        RIP version    : 1
        Preference     : 100
        Checkzero      : Enabled
        Default-cost   : 0
        Auto Summary   : Enabled
        Host-route     : Enabled
......
```

可以观察到默认开启了自动汇总。

3. 配置 RIPv2 自动汇总

在路由器 R1、R2、R3 上配置 **version 2** 命令，运行 RIPv2 协议。

```
[R1]rip 1
[R1-rip-1]version 2

[R2]rip 1
[R2-rip-1]version 2

[R3]rip 1
[R3-rip-1]version 2
```

配置完成后，在 R3 的 S 1/0/1 接口上抓包，如图 7-7 所示。

No.	Time	Source	Destination	Protocol	Length	TX rate	RSSI	Frequency/channel	Info
982	2090.242000000	192.168.23.3	224.0.0.9	RIPv2	116				Response
991	2110.195000000	192.168.23.2	224.0.0.9	RIPv2	56				Response
996	2121.474000000	192.168.23.3	224.0.0.9	RIPv2	116				Response
1005	2139.367000000	192.168.23.2	224.0.0.9	RIPv2	56				Response

```
⊞ Frame 996: 116 bytes on wire (928 bits), 116 bytes captured (928 bits) on interface 0
⊞ Point-to-Point Protocol
⊞ Internet Protocol Version 4, Src: 192.168.23.3 (192.168.23.3), Dst: 224.0.0.9 (224.0.0.9)
⊞ User Datagram Protocol, Src Port: router (520), Dst Port: router (520)
⊟ Routing Information Protocol
    Command: Response (2)
    Version: RIPv2 (2)
  ⊞ IP Address: 3.3.0.0, Metric: 1
  ⊟ IP Address: 3.3.1.0, Metric: 1
      Address Family: IP (2)
      Route Tag: 0
      IP Address: 3.3.1.0 (3.3.1.0)
      Netmask: 255.255.255.0 (255.255.255.0)
      Next Hop: 0.0.0.0 (0.0.0.0)
      Metric: 1
  ⊞ IP Address: 3.3.2.0, Metric: 1
  ⊞ IP Address: 3.3.3.0, Metric: 1
```

图 7-7 抓包观察

可以观察到，RIPv2 报文中携带了掩码信息。RIPv2 支持自动汇总，默认是开启的，并且可以关闭。

查看 R1 与 R2 的路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 10        Routes : 10
Destination/Mask    Proto   Pre  Cost    Flags  NextHop         Interface
      3.3.0.0/24    RIP     100  2       D      192.168.12.2    Serial1/0/0
      3.3.1.0/24    RIP     100  2       D      192.168.12.2    Serial1/0/0
      3.3.2.0/24    RIP     100  2       D      192.168.12.2    Serial1/0/0
      3.3.3.0/24    RIP     100  2       D      192.168.12.2    Serial1/0/0
```

```
    127.0.0.0/8    Direct  0    0     D    127.0.0.1       InLoopBack0
......

[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 12        Routes : 12
Destination/Mask    Proto   Pre  Cost    Flags   NextHop         Interface
    3.3.0.0/24     RIP     100  1       D      192.168.23.3   Serial1/0/0
    3.3.1.0/24     RIP     100  1       D      192.168.23.3   Serial1/0/0
    3.3.2.0/24     RIP     100  1       D      192.168.23.3   Serial1/0/0
    3.3.3.0/24     RIP     100  1       D      192.168.23.3   Serial1/0/0
    127.0.0.0/8    Direct  0    0       D      127.0.0.1      InLoopBack0
......
```

可以观察到，接收到的路由条目是具体的明细路由条目，而没有汇总路由，即此时RIPv2默认自动汇总并没有生效。

这是因为在华为设备上，以太网接口和串口都默认启用了水平分割功能。为了防止环路和不连续子网问题的产生，在启用了水平分割或毒性逆转的接口上，RIPv2 的默认自动汇总就会失效，所以从 R3 通告过来的都是具体的明细路由条目。

要使 RIPv2 的默认自动汇总生效，有两种方法。

第一种方法，使用 **summary always** 命令。配置该命令后，不论水平分割是否启用，RIPv2 的自动汇总都生效。

```
[R3]rip
[R3-rip-1]version 2
[R3-rip-1]summary always
```

第二种方法，关闭相应接口下的水平分割功能。

```
[R3]interface Serial 1/0/1
[R3-Serial1/0/1]undo rip split-horizon
```

使用以上的任一种方法后，查看 R1 与 R2 的路由表。

```
<R1>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7         Routes : 7
Destination/Mask    Proto   Pre   Cost     Flags    NextHop          Interface
    3.0.0.0/8      RIP     100   2        D       192.168.12.2    Serial1/0/0
    127.0.0.0/8    Direct  0     0        D       127.0.0.1       InLoopBack0
......

<R2>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 9         Routes : 9
Destination/Mask    Proto   Pre   Cost     Flags    NextHop          Interface
    3.0.0.0/8      RIP     100   1        D       192.168.23.3    Serial1/0/0
    127.0.0.0/8    Direct  0     0        D       127.0.0.1       InLoopBack0
......
```

可以观察到，此时 RIPv2 的自动汇总生效了。

4. 配置 RIPv2 手动汇总

配置手动汇总需首先删除上一步骤中使 RIPv2 自动汇总功能生效的配置，这里省略此步骤。

在 R3 上使用 **rip summary-address** 命令配置手动汇总，配合需要汇总的本地网络 IP 地址为 3.3.0.0，网络掩码为 255.255.252.0。

```
[R3]interface Serial 1/0/1
[R3-Serial1/0/1]rip summary-address 3.3.0.0 255.255.252.0
```

配置完成后，查看 R1 与 R2 的路由表。

```
<R1>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7          Routes : 7
Destination/Mask    Proto   Pre  Cost       Flags  NextHop         Interface
       3.3.0.0/22   RIP     100  2            D    192.168.12.2    Serial1/0/0
     127.0.0.0/8    Direct  0    0            D    127.0.0.1       InLoopBack0
     127.0.0.1/32   Direct  0    0            D    127.0.0.1       InLoopBack0
  192.168.12.0/24   Direct  0    0            D    192.168.12.1    Serial1/0/0
  192.168.12.1/32   Direct  0    0            D    127.0.0.1       Serial1/0/0
  192.168.12.2/32   Direct  0    0            D    192.168.12.2    Serial1/0/0
  192.168.23.0/24   RIP     100  1            D    192.168.12.2    Serial1/0/0

<R2>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 9          Routes : 9
Destination/Mask    Proto   Pre  Cost      Flags  NextHop         Interface
       3.3.0.0/22   RIP     100  1           D    192.168.23.3    Serial1/0/0
     127.0.0.0/8    Direct  0    0           D    127.0.0.1       InLoopBack0
     127.0.0.1/32   Direct  0    0           D    127.0.0.1       InLoopBack0
  192.168.12.0/24   Direct  0    0           D    192.168.12.2    Serial1/0/1
  192.168.12.1/32   Direct  0    0           D    192.168.12.1    Serial1/0/1
  192.168.12.2/32   Direct  0    0           D    127.0.0.1       Serial1/0/1
  192.168.23.0/24   Direct  0    0           D    192.168.23.2    Serial1/0/0
  192.168.23.2/32   Direct  0    0           D    127.0.0.1       Serial1/0/0
  192.168.23.3/32   Direct  0    0           D    192.168.23.3    Serial1/0/0
```

可以观察到，R1 和 R2 上已经接收到了该汇总路由条目，且没有任何明细路由条目。

思考

华为设备默认开启了 RIPv2 的自动汇总，如果没有默认开启接口下的水平分割，即自动汇总生效的情况下，可能会导致出现环路以及不连续子网等问题。请设计一个相关场景，模拟在 RIPv2 开启了自动汇总且关闭了水平分割的情况下，导致路由环路或不连续子网问题的出现。

7.4　配置 RIP 的版本兼容、定时器及协议优先级

原理概述

RIP 在 IPv4 中有 v1 和 v2 两个版本。在配置 RIP 时，如果不指定版本，接口默认情况下能接收 v1 和 v2 的报文，但只能发送 v1 的报文；在指定版本的情况下，RIPv1 只能接收和发送 v1 的报文，RIPv2 只能接收和发送 v2 的报文。

RIP 的定时器有 3 种：更新计时器，默认每 30s 发送一次更新；超时计时器，默认时间 180s，如果在超时计时器内没有收到邻居发来的更新报文，则把该路由的度量值设置为 16，并启动垃圾收集定时器；垃圾收集定时器，默认时间 120s，如果启动了该计时器，那么 120s 超时以后，路由表中会删除该路由表项。

RIP 默认协议优先级为 100，可以手动修改。

实验内容

本实验中采用简单的场景介绍 RIP 各版本间的区别及如何实现相互间的兼容、RIP 的 3 种定时器的作用及修改方法、RIP 优先级的作用及修改方法。

实验目的

- 掌握配置 RIP 版本的方法
- 理解 RIPv1 和 RIPv2 的相互兼容性
- 掌握 RIP 的 3 种定时器的配置
- 掌握 RIP 的协议优先级的配置

实验拓扑

配置 RIP 的版本兼容、定时器及协议优先级的拓扑如图 7-8 所示。

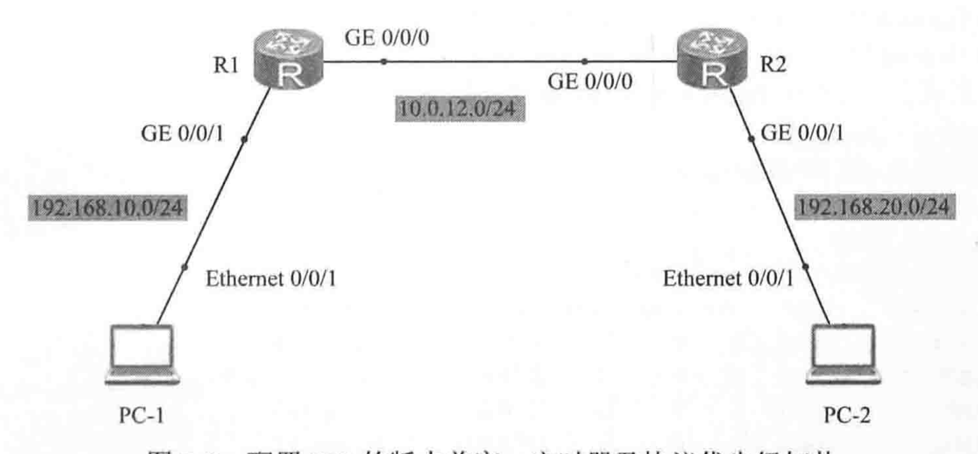


图 7-8　配置 RIP 的版本兼容、定时器及协议优先级拓扑

实验编址

实验编址见表 7-4。

表 7-4 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR2220）	GE 0/0/0	10.0.12.1	255.255.255.0	N/A
	GE 0/0/1	192.168.10.1	255.255.255.0	N/A
R2（AR2220）	GE 0/0/0	10.0.12.2	255.255.255.0	N/A
	GE 0/0/1	192.168.20.1	255.255.255.0	N/A
PC-1	Ethernet 0/0/1	192.168.10.10	255.255.255.0	192.168.10.1
PC-2	Ethernet 0/0/1	192.168.20.10	255.255.255.0	192.168.20.1

实验步骤

1. 基本配置

根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测各直连链路的连通性。

```
[R1]ping -c 1 10.0.12.2
  PING 10.0.12.2: 56   data bytes, press CTRL_C to break
 Reply from 10.0.12.2: bytes=56 Sequence=1 ttl=255 time=50 ms
  --- 10.0.12.2 ping statistics ---
    1 packet(s) transmitted
    1 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 50/50/50 ms
```

其余直连网段的连通性测试省略。

2. 配置 RIP 协议的版本兼容

按照图 7-8 分别在 R1 和 R2 上配置 RIP 协议，通告相应网段。但是在 R1 上，不指定 RIP 的版本，在 R2 上指定使用版本 v2。

```
[R1]rip
[R1-rip-1]network 10.0.0.0
[R1-rip-1]network 192.168.10.0

[R2]rip
[R2-rip-1]network 10.0.0.0
[R2-rip-1]network 192.168.20.0
[R2-rip-1]version 2
```

配置完成后，使用 **display ip routing-table** 命令查看 R1 和 R2 的路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7          Routes : 7
Destination/Mask    Proto   Pre  Cost  Flags  NextHop        Interface
10.0.12.0/24        Direct  0    0     D      10.0.12.1      GigabitEthernet0/0/0
10.0.12.1/32        Direct  0    0     D      127.0.0.1      GigabitEthernet0/0/0
127.0.0.0/8         Direct  0    0     D      127.0.0.1      InLoopBack0
127.0.0.1/32        Direct  0    0     D      127.0.0.1      InLoopBack0
192.168.10.0/24     Direct  0    0     D      192.168.10.1   GigabitEthernet0/0/1
192.168.10.1/32     Direct  0    0     D      127.0.0.1      GigabitEthernet0/0/1
192.168.20.0/24     RIP     100  1     D      10.0.12.2      GigabitEthernet0/0/0

[R2]display ip routing-table
```

```
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 6        Routes : 6
Destination/Mask    Proto   Pre  Cost   Flags   NextHop         Interface
10.0.12.0/24        Direct  0    0      D       10.0.12.2       GigabitEthernet0/0/0
10.0.12.2/32        Direct  0    0      D       127.0.0.1       GigabitEthernet0/0/0
127.0.0.0/8         Direct  0    0      D       127.0.0.1       InLoopBack0
127.0.0.1/32        Direct  0    0      D       127.0.0.1       InLoopBack0
192.168.20.0/24     Direct  0    0      D       192.168.20.1    GigabitEthernet0/0/1
192.168.20.1/32     Direct  0    0      D       127.0.0.1       GigabitEthernet0/0/1
```

可以观察到，在R1的路由表中存在PC-2所在网段的路由条目，在R2的路由表中没有发现PC-2所在网段的路由条目。

在R1的GE 0/0/0接口上抓取R1发送给R2和从R2接收到的RIP报文，如图7-9所示。

No.	Time	Source	Destination	Protocol	Info
1	0.000000	10.0.12.2	224.0.0.9	RIPv2	Response
2	10.187000	10.0.12.1	255.255.255.255	RIPv1	Response
3	34.945000	10.0.12.2	224.0.0.9	RIPv2	Response
4	40.217000	10.0.12.1	255.255.255.255	RIPv1	Response

```
⊞ Frame 4: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)
⊞ Ethernet II, Src: HuaweiTe_03:29:72 (00:e0:fc:03:29:72), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
⊞ Internet Protocol, Src: 10.0.12.1 (10.0.12.1), Dst: 255.255.255.255 (255.255.255.255)
⊞ User Datagram Protocol, Src Port: router (520), Dst Port: router (520)
⊟ Routing Information Protocol
    Command: Response (2)
    Version: RIPv1 (1)
  ⊟ IP Address: 192.168.10.0, Metric: 1
      Address Family: IP (2)
      IP Address: 192.168.10.0 (192.168.10.0)
      Metric: 1
```

图7-9 抓包观察

可以观察到R1采用版本1，即广播方式来发送更新；而R2采用版本2，即组播方式发送更新。验证了R1在RIP协议进程中没有明确指定版本配置时，默认是可以处理接收版本1和版本2的报文，但仅发送版本1的报文；而R2因在RIP协议进程中明确配置了版本 2，仅接收和发送版本2的报文。

因此，由于R1发送的是RIPv1报文，而R2不能正确处理接收，所以R2的路由表中没有PC-1所在网段的路由条目。而R2发送的RIPv2报文能够被R1处理接收，所以在R1的路由表中存在PC-2所在网段的路由条目。

现在为了使R2也能接收PC-1所在网段的路由条目，在R1上设置接口的RIP版本，使R1能够以广播方式发送RIPv2报文。

```
[R1]interface GigabitEthernet0/0/0
[R1-GigabitEthernet0/0/0]rip version 2 broadcast
```

配置完成后，查看R2的路由表。

```
[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7        Routes : 7
Destination/Mask    Proto   Pre  Cost   Flags   NextHop         Interface
```

```
10.0.12.0/24        Direct   0     0    D    10.0.12.2       GigabitEthernet0/0/0
10.0.12.2/32        Direct   0     0    D    127.0.0.1       GigabitEthernet0/0/0
127.0.0.0/8         Direct   0     0    D    127.0.0.1       InLoopBack0
127.0.0.1/32        Direct   0     0    D    127.0.0.1       InLoopBack0
192.168.10.0/24     RIP      100   1    D    10.0.12.1       GigabitEthernet0/0/0
192.168.20.0/24     Direct   0     0    D    192.168.20.1    GigabitEthernet0/0/1
192.168.20.1/32     Direct   0     0    D    127.0.0.1       GigabitEthernet0/0/1
```

发现路由表中已经存在 R1 发送过来的路由条目。同样也可以使用 **rip version 2 mutlicast** 命令，即使 R1 能够以组播方式发送 RIPv2 报文，效果一样，这里不再验证。

在配置 RIP 协议时建议路由器之间配置相同 RIP 版本，即所有路由器都配置 RIPv1 或者都配置 RIP v2，以避免可能由于错误配置而导致 RIP 协议无法正常工作。

3. 配置 RIP 的定时器

配置完 RIP 版本兼容后，再次在 R1 的 GE 0/0/0 接口上通过抓包分析 R1 和 R2 更新报文的发送情况，如图 7-10 所示。

2	10.187000	10.0.12.1
3	34.945000	10.0.12.2
4	40.217000	10.0.12.1
5	65.942000	10.0.12.2

图 7-10 抓包观察

可以观察到 R1 在 10s 时发送了一次更新，R2 在 34s 时发送了一次更新，R1 在 40s 时发送了下一次更新，R2 在 65s 时发送了下一次更新。即默认情况下 RIP 协议会每隔 30s 左右发送一次路由更新。

路由更新的有效期为超时定时器定义的时间 180s。即当在 180s 内没有收到新的路由更新，则宣布该路由不可达，并从路由表中清除掉该路由条目。

为了验证效果，在 R1 的 GE 0/0/0 接口上配置停止发送 RIP 路由更新。

```
[R1]interface GigabitEthernet0/0/0
[R1-GigabitEthernet0/0/0]undo rip output
```

配置完成后，此时 R1 的 GE 0/0/0 接口上已经无法发送任何 RIP 路由更新，此时立刻查看 R2 的路由表。

```
[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7        Routes : 7
Destination/Mask    Proto    Pre   Cost      Flags   NextHop         Interface
10.0.12.0/24        Direct   0     0         D       10.0.12.2       GigabitEthernet0/0/0
10.0.12.2/32        Direct   0     0         D       127.0.0.1       GigabitEthernet0/0/0
127.0.0.0/8         Direct   0     0         D       127.0.0.1       InLoopBack0
127.0.0.1/32        Direct   0     0         D       127.0.0.1       InLoopBack0
192.168.10.0/24     RIP      100   1         D       10.0.12.1       GigabitEthernet0/0/0
192.168.20.0/24     Direct   0     0         D       192.168.20.1    GigabitEthernet0/0/1
192.168.20.1/32     Direct   0     0         D       127.0.0.1       GigabitEthernet0/0/1
```

可以观察到，从 R1 接收到的路由条目依然存在，原因是 RIP 超时定时器没有到期，该路由条目依然被保存在路由表中。

使用 **display rip database** 命令检查 R2 的 RIP 发布数据库中的所有激活路由。

```
[R2]display rip 1 database
Advertisement State : [A] - Advertised
[I] - Not Advertised/Withdraw
---------------------------------------------------------------------------
    10.0.0.0/8, cost 0, ClassfulSumm
    10.0.12.0/24, cost 0, [A], Rip-interface
    192.168.10.0/24, cost 1, ClassfulSumm
192.168.10.0/24, cost 1, [A], nexthop 10.0.12.1
    192.168.20.0/24, cost 0, ClassfulSumm
192.168.20.0/24, cost 0, [A], Rip-interface
```

路由条目也没有发生变化，状态仍然为[A]，即仍被通告。在等待超时计时器到期定义的 180s 以后再使用 display ip routing-table 命令检查。

```
[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
---------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7        Routes : 7
Destination/Mask    Proto   Pre  Cost   Flags NextHop        Interface
10.0.12.0/24        Direct  0    0      D     10.0.12.2      GigabitEthernet0/0/0
10.0.12.2/32        Direct  0    0      D     127.0.0.1      GigabitEthernet0/0/0
127.0.0.0/8         Direct  0    0      D     127.0.0.1      InLoopBack0
127.0.0.1/32        Direct  0    0      D     127.0.0.1      InLoopBack0
192.168.20.0/24     Direct  0    0      D     192.168.20.1   GigabitEthernet0/0/1
192.168.20.1/32     Direct  0    0      D     127.0.0.1      GigabitEthernet0/0/1
```

可以观察到，R2 的路由表中已经无法看到 R1 发送过来的路由条目，原因是超时定时器已经到期，该路由条目被定义为失效，已经从路由表中清除了。

同时再次检查 R2 的路由表和发布数据库。

```
[R2]display rip 1 database
Advertisement State : [A] - Advertised
[I] - Not Advertised/Withdraw
---------------------------------------------------------------------------
    10.0.0.0/8, cost 0, ClassfulSumm
    10.0.12.0/24, cost 0, [A], Rip-interface
    192.168.10.0/24, cost 16, ClassfulSumm
    192.168.10.0/24, cost 16, [I], nexthop 10.0.12.1
    192.168.20.0/24, cost 0, ClassfulSumm
    192.168.20.0/24, cost 0, [A], Rip-interface
```

发现在数据库中可以看到该路由条目，但是该路由条目已经被标记为 16 跳，即不可达，并且状态标记为[I]，该路由将不能被通告出去。虽然该条目已失效，但是仍然存在于发布数据库中的原因是 RIP 垃圾收集定时器启动，且还没有到期，暂时不能从数据库中清除。

如果在默认 120s 内仍然没有收到更新报文，垃圾收集定时器超时后将删除该表项。经过 120s 后再查看 R2 上的发布数据库。

```
[R2]display rip 1 database
---------------------------------------------------------------------------
 Advertisement State : [A] - Advertised
                               [I] - Not Advertised/Withdraw
---------------------------------------------------------------------------
   10.0.0.0/8, cost 0, ClassfulSumm
```

```
10.0.12.0/24, cost 0, [A], Rip-interface
192.168.20.0/24, cost 0, ClassfulSumm
192.168.20.0/24, cost 0, [A], Rip-interface
```

可以观察到，此时已经不存在任何 R1 发送过来的路由条目。

可以通过 **timers rip** 命令改变这几个定时器的默认值来影响 RIP 的收敛速度。现将 R1 的更新报文的时间间隔修改为 20s，超时计时器的超时时间修改为 120s，垃圾收集计时器的超时时间修改为 60s。

```
[R1]rip 1
[R1-rip-1]timers rip 20 120 60
```

配置完成后，查看 RIP 的协议信息。

```
[R1]display rip
Public VPN-instance
        RIP process : 1
        RIP version    : 2
        Preference      : 100
        Checkzero       : Enabled
        Default-cost    : 0
        Summary          : Enabled
        Host-route      : Enabled
        Maximum number of balanced paths : 32
       Update time     : 20 s                  Age time : 120 s
       Garbage-collect time : 60 s
        Graceful restart   : Disabled
        BFD                    : Disabled
......
```

可以观察到，RIP 定时器的值在更改后立即生效。

如果 3 个定时器值设置不当，会引起网络不稳定。例如，如果更新时间大于失效时间，那么在更新时间内，可能在接收到路由更新之前，本地的路由条目已经失效了。定时器值的调整应考虑网络的规模和性能，并在所有运行的 RIP 路由器上进行统一配置。

4. 配置 RIP 协议优先级

在实际网络中，去往相同目的网段的路由信息可以通过不同的路由协议获取，比如同时通过静态路由和 RIP 协议获取，此时就会先比较二者的协议优先级，通过具有较高优先级的路由协议所获取的路由信息将被优选放入路由表中。

查看 R1 的路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
       Destinations : 7          Routes : 7
Destination/Mask    Proto    Pre    Cost    Flags    NextHop         Interface
10.0.12.0/24        Direct   0      0       D        10.0.12.1       GigabitEthernet0/0/0
10.0.12.1/32        Direct   0      0       D        127.0.0.1       GigabitEthernet0/0/0
127.0.0.0/8         Direct   0      0       D        127.0.0.1       InLoopBack0
127.0.0.1/32        Direct   0      0       D        127.0.0.1       InLoopBack0
192.168.10.0/24     Direct   0      0       D        192.168.10.1    GigabitEthernet0/0/1
192.168.10.1/32     Direct   0      0       D        127.0.0.1       GigabitEthernet0/0/1
192.168.20.0/24     RIP      100    1       D        10.0.12.2       GigabitEthernet0/0/0
```

可以观察到 RIP 的路由优先级默认值为 100。可以使用 **preference** 命令将 R1 的路由

优先级调整为 90，然后查看 R1 的路由表。

```
[R1]rip
[R1-rip-1]preference 90

[R1-rip-1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
Destinations : 7        Routes : 7
Destination/Mask    Proto   Pre  Cost   Flags  NextHop        Interface
10.0.12.0/24        Direct  0    0      D      10.0.12.1      GigabitEthernet0/0/0
10.0.12.1/32        Direct  0    0      D      127.0.0.1      GigabitEthernet0/0/0
127.0.0.0/8         Direct  0    0      D      127.0.0.1      InLoopBack0
127.0.0.1/32        Direct  0    0      D      127.0.0.1      InLoopBack0
192.168.10.0/24     Direct  0    0      D      192.168.10.1   GigabitEthernet0/0/1
192.168.10.1/32     Direct  0    0      D      127.0.0.1      GigabitEthernet0/0/1
192.168.20.0/24     RIP     90   1      D      10.0.12.2      GigabitEthernet0/0/0
```

可以观察到此时已经完成了相应修改。注意优先级的数值越小，代表优先级越高。

思考

在此实验中，如果在 R1 上配置一条去往 192.168.20.0 网段的静态路由，再把 RIP 优先级修改为60，那么在R1的IP路由表中该网段路由来自RIP还是静态路由？为什么？

7.5　配置 RIP 抑制接口及单播更新

原理概述

RIP 支持抑制接口的配置，即配置后禁止接口发送更新报文，但此接口所在网段的路由可以发布出去。可通过两种方法来实现，执行 **silent-interface** 命令或在接口下配置 **undo rip output** 使其只接收报文，但不能发送 RIP 报文。**silent-interface** 的优先级大于在接口下配置的 **undo rip output**，默认情况下为不抑制状态。还可以在接口下配置 **undo rip input** 命令，禁止接口接收 RIP 更新报文，这也是预防路由环路的一种方式。

单播更新是指 RIP 使用单播发送 RIP 报文。在默认情况下，RIP 每隔 30s 以广播或组播方式交换整个路由表的信息，这将耗费大量网络带宽，特别是在广域网中，可能出现严重性能问题。为了解决因 RIP 的广播报文而产生的网络性能问题，可以使用单播更新的方式来交换路由信息。当使用 **silent-interface** 命令配置抑制接口后，再指定单播更新的目的地址后，单播更新有效；如果在接口下使用 **undo rip output** 命令来配置抑制接口，即使再指定单播更新的目的地址也是无法发送更新的路由条目的。

RIPv1 和 RIPv2 对于抑制接口和单播更新的特性支持情况相同。

实验目的

- 掌握 RIP 抑制接口的配置

- 理解抑制接口的原理及应用场景
- 掌握 RIP 中单播更新的配置
- 理解单播更新的原理及应用场景

实验内容

本实验模拟企业网络场景。R1 为该公司出口网关路由器，连接运营商网络；R2 为公司 IT 部门路由器，通过交换机 S1 与网关相连；人事部员工直接通过交换机 S1 接入公司网络；R3 为公司财务部门路由器，同样通过 S1 与网关相连。所有路由器运行路由协议 RIP 实现网络互通。由于交换机 S1 直连了大量 PC 用户，如果 R1 继续以广播（RIPv1）或组播（RIPv2）的方式发送更新的路由给 R2 和 R3，处于同一广播网络中的 S1 下连接的 PC 也会收到这些对 PC 来说无用的更新，造成了带宽和资源的浪费。为了优化网络，现需在 R1 的 GE 0/0/1 接口配置抑制接口来抑制广播或组播更新，为了使 R2 和 R3 能照常接收更新，还需要在 R1 上配置与 R2、R3 的单播更新；同时禁止其他部门访问财务部门，需抑制 R3 的 E 1/0/1 接口，不发布任何 RIP 路由（单播更新也不行），仅可接收其他路由信息。

实验拓扑

配置 RIP 抑制接口及单播更新的拓扑如图 7-11 所示。

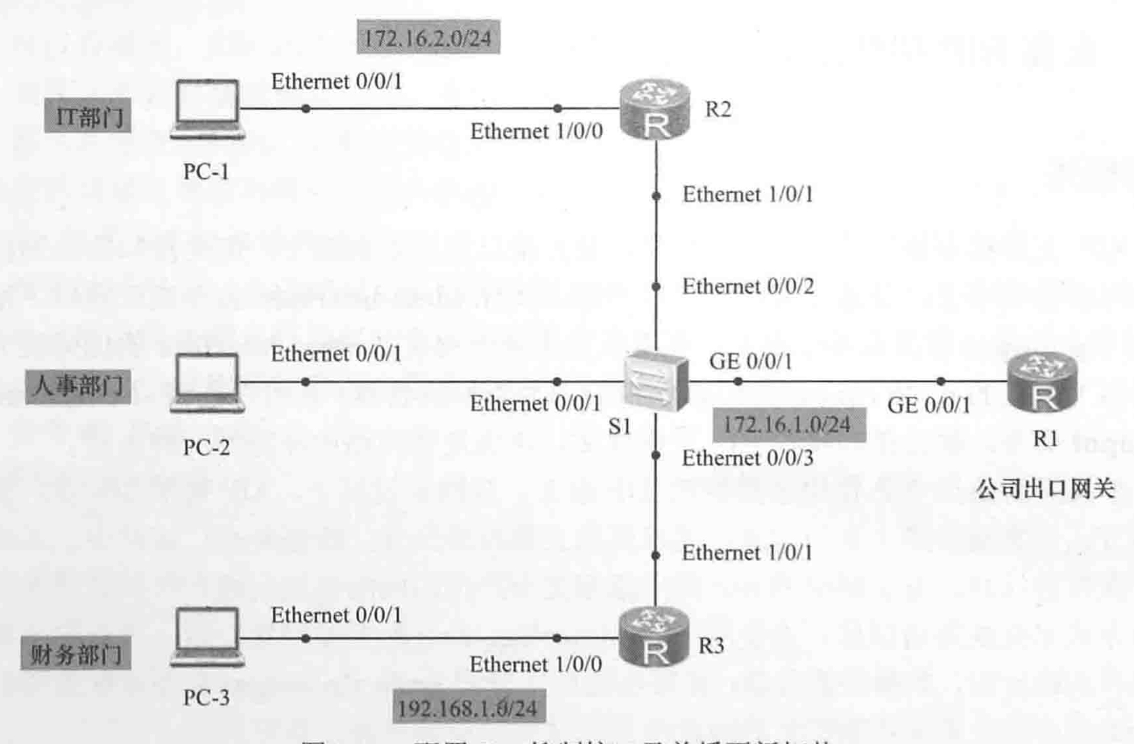


图 7-11　配置 RIP 抑制接口及单播更新拓扑

实验编址

实验编址见表 7-5。

表 7-5 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR2220）	GE 0/0/1	172.16.1.254	255.255.255.0	N/A
R2（AR2220）	Ethernet 1/0/1	172.16.1.100	255.255.255.0	N/A
	Ethernet 1/0/0	172.16.2.254	255.255.255.0	N/A
R3（AR2220）	Ethernet 1/0/1	172.16.1.200	255.255.255.0	N/A
	Ethernet 1/0/0	192.168.1.254	255.255.255.0	N/A
PC-1	Ethernet 0/0/1	172.16.2.1	255.255.255.0	172.16.2.254
PC-2	Ethernet 0/0/1	172.16.1.1	255.255.255.0	172.16.1.254
PC-3	Ethernet 0/0/1	192.168.1.1	255.255.255.0	192.168.1.254

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 地址配置，并使用 **ping** 命令检测各直连链路的连通性。

```
[R1]ping -c 1 172.16.1.1
  PING 172.16.1.1: 56   data bytes, press CTRL_C to break
    Reply from 172.16.1.1: bytes=56 Sequence=1 ttl=128 time=50 ms
  --- 172.16.1.1 ping statistics ---
    1 packet(s) transmitted
    1 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 50/50/50 ms
```

其余直连网段的连通性测试省略。

2. 搭建基础的 RIP 网络

在公司各台路由器上都运行 RIP 路由协议，并通告相应网段。

```
[R1]rip 1
[R1-rip-1]network 172.16.0.0

[R2]rip 1
[R2-rip-1]network 172.16.0.0

[R3]rip 1
[R3-rip-1]network 172.16.0.0
[R3-rip-1]network 192.168.1.0
```

配置完成后，检查 3 台设备的路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 6        Routes : 6
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
    127.0.0.0/8     Direct  0    0           D   127.0.0.1       InLoopBack0
    127.0.0.1/32    Direct  0    0           D   127.0.0.1       InLoopBack0
    172.16.1.0/24   Direct  0    0           D   172.16.1.254    GigabitEthernet0/0/1
    172.16.1.254/32 Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/1
    172.16.2.0/24   RIP     100  1           D   172.16.1.100    GigabitEthernet0/0/1
    192.168.1.0/24  RIP     100  1           D   172.16.1.200    GigabitEthernet0/0/1
```

```
[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7        Routes : 7
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
   127.0.0.0/8      Direct  0    0           D   127.0.0.1       InLoopBack0
   127.0.0.1/32     Direct  0    0           D   127.0.0.1       InLoopBack0
   172.16.1.0/24    Direct  0    0           D   172.16.1.100    Ethernet1/0/1
   172.16.1.100/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/1
   172.16.2.0/24    Direct  0    0           D   172.16.2.254    Ethernet1/0/0
   172.16.2.254/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/0
   192.168.1.0/24   RIP     100  1           D   172.16.1.200    Ethernet1/0/1

[R3]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7        Routes : 7
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
   127.0.0.0/8      Direct  0    0           D   127.0.0.1       InLoopBack0
   127.0.0.1/32     Direct  0    0           D   127.0.0.1       InLoopBack0
   172.16.1.0/24    Direct  0    0           D   172.16.1.200    Ethernet1/0/1
   172.16.1.200/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/1
   172.16.2.0/24    RIP     100  1           D   172.16.1.100    Ethernet1/0/1
   192.168.1.0/24   Direct  0    0           D   192.168.1.254   Ethernet1/0/0
   192.168.1.254/32 Direct  0    0           D   127.0.0.1       Ethernet1/0/0
```

可以观察到此时每台路由器上都已经拥有了所有网段的路由信息，连通性检查这里省略。

接下来网络管理员在 PC-2 的 E 0/0/1 接口上抓包，如图 7-12 所示，可以观察到接收了许多对 PC-2 来说无用的 RIP 路由更新。

No.	Time	Source	Destination	Protocol	Length	Info
373	784.825000	HuaweiTe_50:ba:4b	Spanning-tree-(for-STP		119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
374	787.067000	HuaweiTe_50:ba:4b	Spanning-tree-(for-STP		119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
375	789.302000	HuaweiTe_50:ba:4b	Spanning-tree-(for-STP		119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
376	791.549000	HuaweiTe_50:ba:4b	Spanning-tree-(for-STP		119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
377	793.748000	HuaweiTe_50:ba:4b	Spanning-tree-(for-STP		119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
378	795.832000	HuaweiTe_50:ba:4b	Spanning-tree-(for-STP		119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
379	796.759000	172.16.1.200	255.255.255.255	RIPv1	66	Response
380	798.116000	HuaweiTe_50:ba:4b	Spanning-tree-(for-STP		119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
381	800.316000	HuaweiTe_50:ba:4b	Spanning-tree-(for-STP		119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
382	802.500000	HuaweiTe_50:ba:4b	Spanning-tree-(for-STP		119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
383	802.999000	172.16.1.254	255.255.255.255	RIPv1	66	Response
384	804.700000	HuaweiTe_50:ba:4b	Spanning-tree-(for-STP		119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
385	806.962000	HuaweiTe_50:ba:4b	Spanning-tree-(for-STP		119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
386	807.086000	172.16.1.100	255.255.255.255	RIPv1	86	Response
387	809.172000	HuaweiTe_50:ba:4b	Spanning-tree-(for-STP		119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
388	811.345000	HuaweiTe_50:ba:4b	Spanning-tree-(for-STP		119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
389	813.498000	HuaweiTe_50:ba:4b	Spanning-tree-(for-STP		119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
390	815.666000	HuaweiTe_50:ba:4b	Spanning-tree-(for-STP		119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
391	817.835000	HuaweiTe_50:ba:4b	Spanning-tree-(for-STP		119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por

图 7-12　抓包观察

在 PC-1 的 E 0/0/1 接口下抓包，如图 7-13 所示。

No.	Time	Source	Destination	Protocol	Length	Info
27	301.6430000	172.16.2.254	255.255.255.255	RIPv1	86	Response
28	335.6510000	172.16.2.254	255.255.255.255	RIPv1	106	Response
29	370.6580000	172.16.2.254	255.255.255.255	RIPv1	106	Response
30	398.7070000	172.16.2.254	255.255.255.255	RIPv1	106	Response
31	428.6900000	172.16.2.254	255.255.255.255	RIPv1	106	Response
32	454.7110000	172.16.2.254	255.255.255.255	RIPv1	106	Response
33	487.7210000	172.16.2.254	255.255.255.255	RIPv1	106	Response
34	520.7150000	172.16.2.254	255.255.255.255	RIPv1	106	Response
35	548.7180000	172.16.2.254	255.255.255.255	RIPv1	106	Response
36	578.7170000	172.16.2.254	255.255.255.255	RIPv1	106	Response

图 7-13 抓包观察

同样可观察到 PC-1 上接收到许多对 PC-1 而言无用的 RIP 更新，这是由于在 R2 上 172.16.2.0 网段也被通告进了 RIP 协议中，即 R2 的 E 1/0/0 接口运行在了 RIP 协议中，也会发送 RIP 路由信息。在 PC-3 上抓包可以看到同样的效果，原理和 PC-1 一样。

3. 配置 RIP 抑制接口，优化公司网络

为了减少对带宽和资源的浪费，不让人事部门、IT 部门和财务部门的 PC 收到大量无关 RIP 报文，可以采用抑制接口的方法来实现，使得该接口只接收 RIP 更新报文，而不发送更新报文。

在各路由器上使用 **silent-interface** 命令将相应接口配置成为抑制接口。

```
[R1]rip 1
[R1-rip-1]silent-interface GigabitEthernet 0/0/1

[R2]rip 1
[R2-rip-1]silent-interface Ethernet 1/0/1
[R2-rip-1]silent-interface Ethernet 1/0/0

[R3]rip 1
[R3-rip-1]silent-interface Ethernet 1/0/1
[R3-rip-1]silent-interface Ethernet 1/0/0
```

配置完成后，分别在 R1、R2、R3 上使用 **display rip** 命令查看相关配置信息。

```
[R1-rip-1]display rip 1
Public VPN-instance
    RIP process : 1
……
       BFD                        : Disabled
Silent-interfaces :
GigabitEthernet0/0/1
       Default-route : Disabled
……
```

可以观察到，配置已经成功。

配置完成后，在 PC-2 的 E 0/0/1 接口上抓包，如图 7-14 所示。

No.	Time	Source	Destination	Protocol	Length	Info
483	989.280000C	HuaweiTe_50:ba:4b	Spanning-tree-(foi	STP	119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
484	991.526000C	HuaweiTe_50:ba:4b	Spanning-tree-(foi	STP	119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
485	993.726000C	HuaweiTe_50:ba:4b	Spanning-tree-(foi	STP	119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
486	995.879000C	HuaweiTe_50:ba:4b	Spanning-tree-(foi	STP	119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
487	998.110000C	HuaweiTe_50:ba:4b	Spanning-tree-(foi	STP	119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
488	1000.27800C	HuaweiTe_50:ba:4b	Spanning-tree-(foi	STP	119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
489	1002.49300C	HuaweiTe_50:ba:4b	Spanning-tree-(foi	STP	119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
490	1004.59900C	HuaweiTe_50:ba:4b	Spanning-tree-(foi	STP	119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
491	1006.79900C	HuaweiTe_50:ba:4b	Spanning-tree-(foi	STP	119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
492	1008.99900C	HuaweiTe_50:ba:4b	Spanning-tree-(foi	STP	119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
493	1011.16700C	HuaweiTe_50:ba:4b	Spanning-tree-(foi	STP	119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
494	1013.39800C	HuaweiTe_50:ba:4b	Spanning-tree-(foi	STP	119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por
495	1015.56600C	HuaweiTe_50:ba:4b	Spanning-tree-(foi	STP	119	MST. Root = 32768/0/4c:1f:cc:50:ba:4b Cost = 0 Por

图 7-14 抓包观察

可以观察到此时 PC-2 已接收不到 RIP 的路由更新。在 PC-1 和 PC-3 的 E 0/0/1 接口

下抓包，效果也一样。

4. 配置RIP单播更新，恢复网络通信

在上一步骤中，网络管理员完成了对网络的优化，使所有的PC都不再接收与之无关的路由更新。而这时各部门员工却反映无法相互间访问，也无法访问外网。

查看R1的路由表。

```
[R1]display IP routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7        Routes : 7
Destination/Mask    Proto   Pre  Cost  Flags  NextHop         Interface
127.0.0.0/8         Direct  0    0     D      127.0.0.1       InLoopBack0
127.0.0.1/32        Direct  0    0     D      127.0.0.1       InLoopBack0
127.255.255.255/32  Direct  0    0     D      127.0.0.1       InLoopBack0
172.16.1.0/24       Direct  0    0     D      172.16.1.254    GigabitEthernet0/0/1
172.16.1.254/32     Direct  0    0     D      127.0.0.1       GigabitEthernet0/0/1
172.16.1.255/32     Direct  0    0     D      127.0.0.1       GigabitEthernet0/0/1
255.255.255.255/32  Direct  0    0     D      127.0.0.1       InLoopBack0
```

可以发现此时R1获取不到其他网络的RIP路由信息，R2、R3路由表同样也接收不到RIP路由信息。原因是在网络管理员将各路由器的相应接口配置成抑制接口后，接口将无法以广播或组播的方式发送RIP更新报文。

现在为了让RIP网络能够正常通信，可以通过增加RIP的单播更新配置来实现。通过该配置，RIP更新报文会以单播形式发送，而不采用正常的组播或广播的形式。

在R1上使用**peer**命令，后面跟上指定的邻居路由器IP地址，即R2和R3与R1相连的直连链路上的IP地址。

```
[R1]rip 1
[R1-rip-1]peer 172.16.1.100
[R1-rip-1]peer 172.16.1.200
```

配置完成后，使用**display rip**命令在R1上检查RIP协议的信息。

```
[R1-rip-1]display rip 1
Public VPN-instance
RIP process : 1
......
       172.16.0.0
Configured peers :
172.16.1.200          172.16.1.100
       Number of routes in database : 4
......
```

可以观察到邻居IP地址已经配置成功。

同样在R2、R3上使用**peer**命令，配置单播更新。

```
[R2]rip 1
[R2-rip-1]peer 172.16.1.254
[R2-rip-1]peer 172.16.1.200

[R3]rip 1
[R3-rip-1]peer 172.16.1.100
[R3-rip-1]peer 172.16.1.254
```

配置完成后，再次查看R1、R2、R3上的路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 6          Routes : 6
Destination/Mask    Proto   Pre  Cost    Flags  NextHop         Interface
   127.0.0.0/8      Direct  0    0       D      127.0.0.1       InLoopBack0
   127.0.0.1/32     Direct  0    0       D      127.0.0.1       InLoopBack0
   172.16.1.0/24    Direct  0    0       D      172.16.1.254    GigabitEthernet0/0/1
   172.16.1.254/32  Direct  0    0       D      127.0.0.1       GigabitEthernet0/0/1
   172.16.2.0/24    RIP     100  1       D      172.16.1.100    GigabitEthernet0/0/1
   192.168.1.0/24   RIP     100  1       D      172.16.1.200    GigabitEthernet0/0/1

[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7          Routes : 7
Destination/Mask    Proto   Pre  Cost    Flags  NextHop         Interface
   127.0.0.0/8      Direct  0    0       D      127.0.0.1       InLoopBack0
   127.0.0.1/32     Direct  0    0       D      127.0.0.1       InLoopBack0
   172.16.1.0/24    Direct  0    0       D      172.16.1.100    Ethernet1/0/1
   172.16.1.100/32  Direct  0    0       D      127.0.0.1       Ethernet1/0/1
   172.16.2.0/24    Direct  0    0       D      172.16.2.254    Ethernet1/0/0
   172.16.2.254/32  Direct  0    0       D      127.0.0.1       Ethernet1/0/0
   192.168.1.0/24   RIP     100  1       D      172.16.1.200    Ethernet1/0/1

[R3]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7          Routes : 7
Destination/Mask    Proto   Pre  Cost    Flags  NextHop         Interface
   127.0.0.0/8      Direct  0    0       D      127.0.0.1       InLoopBack0
   127.0.0.1/32     Direct  0    0       D      127.0.0.1       InLoopBack0
   172.16.1.0/24    Direct  0    0       D      172.16.1.200    Ethernet1/0/1
   172.16.1.200/32  Direct  0    0       D      127.0.0.1       Ethernet1/0/1
   172.16.2.0/24    RIP     100  1       D      172.16.1.100    Ethernet1/0/1
   192.168.1.0/24   Direct  0    0       D      192.168.1.254   Ethernet1/0/0
   192.168.1.254/32 Direct  0    0       D      127.0.0.1       Ethernet1/0/0
```

可以观察到此时每台路由器上都已经拥有了所有网段的路由信息，连通性检查这里省略。现在网络通信恢复正常，并且所有 PC 也不会接收到任何 RIP 报文。

5. 验证另一种抑制接口方式

RIP 协议中还可以通过使用 **undo rip output** 命令来配置抑制接口，禁止接口发送 RIP 报文。

首先将 R3 上的现有抑制接口和单播更新的配置删除，然后在 R3 上的 E 1/0/1 接口上配置 **undo rip output** 命令，禁止接口发送 RIP 报文。

```
[R3]rip 1
[R3-rip-1]undo silent-interface Ethernet 1/0/1
[R3-rip-1]undo silent-interface Ethernet 1/0/0
[R3-rip-1]undo peer 172.16.1.100
[R3-rip-1]undo peer 172.16.1.254
[R3-rip-1]interface Ethernet1/0/1
[R3-Ethernet1/0/1]undo rip output
```

配置完成后，等待一段时间，查看 R1、R2 路由器的路由表。

```
[R1-rip-1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 5        Routes : 5
Destination/Mask    Proto   Pre  Cost    Flags   NextHop         Interface
 127.0.0.0/8        Direct  0    0       D       127.0.0.1       InLoopBack0
 127.0.0.1/32       Direct  0    0       D       127.0.0.1       InLoopBack0
 172.16.1.0/24      Direct  0    0       D       172.16.1.254    GigabitEthernet0/0/1
 172.16.1.254/32    Direct  0    0       D       127.0.0.1       GigabitEthernet0/0/1
 172.16.2.0/24      RIP     100  1       D       172.16.1.100    GigabitEthernet0/0/1

[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 6        Routes : 6
Destination/Mask    Proto   Pre  Cost    Flags   NextHop         Interface
127.0.0.0/8         Direct  0    0       D       127.0.0.1       InLoopBack0
127.0.0.1/32        Direct  0    0       D       127.0.0.1       InLoopBack0
172.16.1.0/24       Direct  0    0       D       172.16.1.100    Ethernet1/0/1
172.16.1.100/32     Direct  0    0       D       127.0.0.1       Ethernet1/0/1
172.16.2.0/24       Direct  0    0       D       172.16.2.254    Ethernet1/0/0
172.16.2.254/32     Direct  0    0       D       127.0.0.1       Ethernet1/0/0
```

可以观察到并没有 R3 上 192.168.1.0 所在直连网段的路由条目，说明 R3 上的 **undo rip output** 命令已经生效，不再发送任何 RIP 路由更新。

在 R3 上配置与 R1 间的单播更新。

```
[R3]rip 1
[R3-rip-1]peer 172.16.1.254
```

配置完成后，等待一段时间，路由表收敛后再在 R1 上查看路由表。

```
[R1-rip-1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 5        Routes : 5
Destination/Mask    Proto   Pre  Cost    Flags   NextHop         Interface
    127.0.0.0/8     Direct  0    0       D       127.0.0.1       InLoopBack0
    127.0.0.1/32    Direct  0    0       D       127.0.0.1       InLoopBack0
    172.16.1.0/24   Direct  0    0       D       172.16.1.254    GigabitEthernet0/0/1
    172.16.1.254/32 Direct  0    0       D       127.0.0.1       GigabitEthernet0/0/1
    172.16.2.0/24   RIP     100  1       D       172.16.1.100    GigabitEthernet0/0/1
```

可以观察到 R1 上仍然没有 192.168.1.0 的路由条目。由此可以证明使用 **undo rip output** 命令来抑制接口，即使配置了单播更新也是无法再以单播的形式发送路由更新的。而在上一步骤当中，当使用 **silent-interface** 命令配置抑制接口后，再使用 **peer** 命令指定邻居 IP 的单播更新目的地址后，单播更新则生效。

在接口下可以使用 **undo rip output** 命令禁止该接口发送 RIP 报文，也可以使用 **undo rip input** 命令来禁止接口接收 RIP 报文，通过这两条命令可以灵活地控制接口对 RIP 报文的发送和接收（默认情况下是可以接收和发送 RIP 报文）。注意 **silent-interface** 命令的优先级大于 **rip output** 或 **rip input** 命令的优先级。

7.6 RIP 与不连续子网

原理概述

RIP 会在主网边界自动汇总，当汇总发生时，汇总的子网路由在边界处被抑制掉，而仅通告主网路由。如果一台路由器上有两个接口，网段分别为 10.1.1.0/24 和 172.16.1.0/24，那么在这两个网段的主网边界路由器就会自动将这两个网段汇总成 10.0.0.0 和 172.16.0.0，并通告给其他路由器。如果主网的子网不连续，被其他主网所分隔，主网边界的自动汇总就会存在问题。

连续子网是指所相连的子网属于同一主网；不连续子网是指相同主网下的子网被另一主网分隔。

实验目的

- 理解连续子网和不连续子网的概念
- 掌握 RIPv1 中解决不连续子网问题的方法
- 掌握 RIPv2 中解决不连续子网问题的方法
- 理解 RIPv1 与 RIPv2 的区别

实验内容

在某公司的网络整改项目中，原先 R1 和 R5 属于同一主网络 10.0.0.0/8，现被 R2、R3、R4 分离，整网采用了 RIPv1 协议，发现在该子网不连续的环境下通信出现了问题，现需要通过额外的配置来解决这些问题，以保证所有设备能够互通。

实验拓扑

RIP 与不连续子网的拓扑如图 7-15 所示。

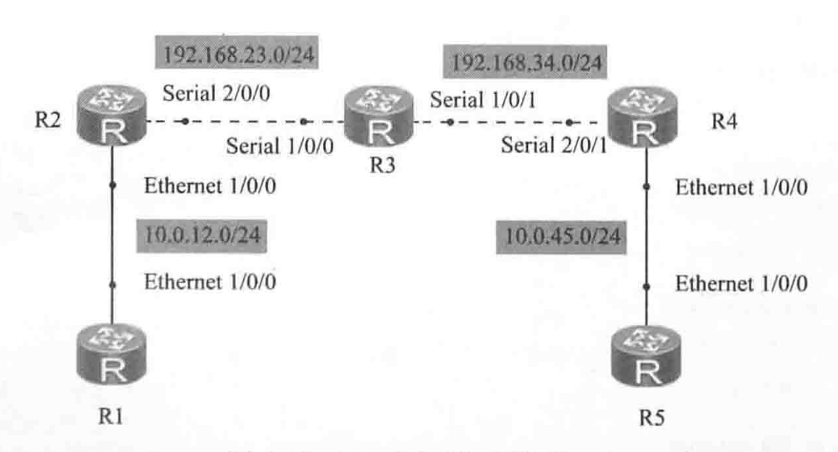


图 7-15 RIP 与不连续子网拓扑

实验编址

实验编址见表 7-6。

表 7-6 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR1220）	Ethernet 1/0/0	10.0.12.1	255.255.255.0	N/A
R2（AR1220）	Ethernet 1/0/0	10.0.12.2	255.255.255.0	N/A
	Serial 2/0/0	192.168.23.2	255.255.255.0	N/A
R3（AR1220）	Serial 1/0/0	192.168.23.3	255.255.255.0	N/A
	Serial 1/0/1	192.168.34.3	255.255.255.0	N/A
R4（AR1220）	Serial 2/0/1	192.168.34.4	255.255.255.0	N/A
	Ethernet 1/0/0	10.0.45.4	255.255.255.0	N/A
R5（AR1220）	Ethernet 1/0/0	10.0.45.5	255.255.255.0	N/A

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 配置，并使用 **ping** 命令测试直连路由器间的连通性。

```
[R1]ping 10.0.12.2
PING 10.0.12.2: 56  data bytes, press CTRL_C to break
    Reply from 10.0.12.2: bytes=56 Sequence=1 ttl=255 time=60 ms
    Reply from 10.0.12.2: bytes=56 Sequence=2 ttl=255 time=20 ms
    Reply from 10.0.12.2: bytes=56 Sequence=3 ttl=255 time=30 ms
    Reply from 10.0.12.2: bytes=56 Sequence=4 ttl=255 time=30 ms
    Reply from 10.0.12.2: bytes=56 Sequence=5 ttl=255 time=50 ms
  --- 10.0.12.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 20/38/60 ms
```

其余直连网段的连通性测试省略。

2. 组建基本的 RIPv1 网络

在路由器 R1、R2、R3、R4、R5 上配置 RIPv1。

```
[R1]rip
[R1-rip-1]network 10.0.0.0

[R2]rip
[R2-rip-1]network 10.0.0.0
[R2-rip-1]network 192.168.23.0

[R3]rip
[R3-rip-1]network 192.168.23.0
[R3-rip-1]network 192.168.34.0

[R4]rip
[R4-rip-1]network 192.168.34.0
[R4-rip-1]network 10.0.0.0

[R5]rip
[R5-rip-1]network 10.0.0.0
```

配置完成后，查看 R1 的路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 6        Routes : 6
Destination/Mask    Proto   Pre  Cost    Flags   NextHop      Interface
    10.0.12.0/24    Direct  0    0       D       10.0.12.1    Ethernet1/0/0
    10.0.12.1/32    Direct  0    0       D       127.0.0.1    Ethernet1/0/0
    127.0.0.0/8     Direct  0    0       D       127.0.0.1    InLoopBack0
    127.0.0.1/32    Direct  0    0       D       127.0.0.1    InLoopBack0
    192.168.23.0/24 RIP     100  1       D       10.0.12.2    Ethernet1/0/0
    192.168.34.0/24 RIP     100  2       D       10.0.12.2    Ethernet1/0/0
```

在 R1 的路由表中，存在 192.168.23.0/24 和 192.168.34.0/24 两条 RIP 路由条目，但并不存在 R4 和 R5 之间的 10.0.45.0/24 路由条目。

查看 R2 的路由表。

```
[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8        Routes : 8
Destination/Mask    Proto   Pre  Cost    Flags   NextHop          Interface
    10.0.12.0/24    Direct  0    0       D       10.0.12.2        Ethernet1/0/0
    10.0.12.2/32    Direct  0    0       D       127.0.0.1        Ethernet1/0/0
    127.0.0.0/8     Direct  0    0       D       127.0.0.1        InLoopBack0
    127.0.0.1/32    Direct  0    0       D       127.0.0.1        InLoopBack0
    192.168.23.0/24 Direct  0    0       D       192.168.23.2     Serial2/0/0
    192.168.23.2/32 Direct  0    0       D       127.0.0.1        Serial2/0/0
    192.168.23.3/32 Direct  0    0       D       192.168.23.3     Serial2/0/0
    192.168.34.0/24 RIP     100  1       D       192.168.23.3     Serial2/0/0
```

在 R2 的路由表中，除 10.0.12.0/24 和 192.168.23.0/24 为直连路由外，仅有 192.168.34.0/24 该条路由条目通过 RIP 接收，却没有 R4 和 R5 之间的 10.0.45.0/24 的路由条目。

查看 R3 的路由表。

```
[R3]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 9        Routes : 10
Destination/Mask    Proto   Pre  Cost    Flags   NextHop          Interface
    10.0.0.0/8      RIP     100  1       D       192.168.23.2     Serial1/0/0
                    RIP     100  1       D       192.168.34.4     Serial1/0/1
    127.0.0.0/8     Direct  0    0       D       127.0.0.1        InLoopBack0
    127.0.0.1/32    Direct  0    0       D       127.0.0.1        InLoopBack0
    192.168.23.0/24 Direct  0    0       D       192.168.23.3     Serial1/0/0
    192.168.23.2/32 Direct  0    0       D       192.168.23.2     Serial1/0/0
    192.168.23.3/32 Direct  0    0       D       127.0.0.1        Serial1/0/0
    192.168.34.0/24 Direct  0    0       D       192.168.34.3     Serial1/0/1
    192.168.34.3/32 Direct  0    0       D       127.0.0.1        Serial1/0/1
    192.168.34.4/32 Direct  0    0       D       192.168.34.4     Serial1/0/1
```

在 R3 的路由表中，除了 192.168.23.0/24 和 192.168.34.0/24 这两个子网是直连之外，分别通过 R2 和 R4 接收到了两条相同的 10.0.0.0/8 的主网路由条目，而并非现网拓扑中

的 10.0.12.0/24 和 10.0.45.0/24 两条子网路由。

导致这种情况的原因是：由于采用了 RIPv1，在 R2 和 R4 分别接收到 10.0.12.0/24 和 10.0.45.0/24 的路由条目时，默认打开了自动有类汇总功能，所以在主网边界向外发送路由信息的时候都汇总成了 10.0.0.0/8，发送给 R3，最终在 R3 上由于接收到了两条目的网段相同、代价值也相同的路由条目。

那么既然此时在 R3 的路由表中存在有 10.0.0.0/8 的路由，在 R3 上测试与 R1 和 R5 的连通性。

```
[R3]ping 10.0.45.5
  PING 10.0.45.5: 56  data bytes, press CTRL_C to break
    Reply from 10.0.45.5: bytes=56 Sequence=1 ttl=254 time=80 ms
    Reply from 10.0.45.5: bytes=56 Sequence=2 ttl=254 time=40 ms
    Reply from 10.0.45.5: bytes=56 Sequence=3 ttl=254 time=80 ms
    Reply from 10.0.45.5: bytes=56 Sequence=4 ttl=254 time=80 ms
    Reply from 10.0.45.5: bytes=56 Sequence=5 ttl=254 time=80 ms
  --- 10.0.45.5 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 40/72/80 ms

[R3]ping 10.0.12.1
  PING 10.0.12.1: 56  data bytes, press CTRL_C to break
    Request time out
    Request time out
    Request time out
    Request time out
    Request time out
  --- 10.0.12.1 ping statistics ---
    5 packet(s) transmitted
    0 packet(s) received
    100.00% packet loss
```

发现此时呈现出有一端无法通信的现象。在 R3 发送 ICMP 报文的时候，会根据路由表进行匹配，即匹配 10.0.0.0/8，那么最终报文流量可能会出现 R3 将本该要发送给 R1 的 ICMP 报文错误地转发给了 R4，导致无法通信。

现在每台设备上的路由表中没有清晰地反馈出拓扑中的真实子网信息，这是由于在 RIPv1 默认自动汇总开启的情况下，设计网络时没有遵循主网的子网应该连续这一要求所致，解决的办法视路由器所使用的 RIP 版本（v1 还是 v2）而有所不同。

3. RIPv1 中解决不连续子网问题

路由器上所运行 RIP 协议的默认版本是 v1，自动汇总无法关闭，所以上面不连续子网所带来的问题，不能通过关闭自动汇总来解决。但如果把不连续的子网转变成连续的子网，问题就可以解决，办法是给接口配置第二个 IP 地址，IP 地址取 10.0.0.0/8 主网的子网。

在路由器 R2 上的 S 2/0/0 接口上配置从 IP 地址，只要在常规配置 IP 地址的命令之后加上 sub 参数即可。

```
[R2]interface serial 2/0/0
[R2-Serial2/0/0]ip address 10.0.23.2 24 sub
```

同理，在R3和R4上也做相应配置，并在R3的RIP进程中添加10.0.0.0网段。

```
[R3]interface serial 1/0/0
[R3-Serial1/0/0]ip address 10.0.23.3 24 sub
[R3-Serial1/0/0]interface serial 0/0/1
[R3-Serial1/0/1]ip address 10.0.34.3 24 sub
[R3-Serial1/0/1]rip
[R3-rip-1]network 10.0.0.0

[R4]interface serial2/0/1
[R4-Serial2/0/1]ip address 10.0.34.4 24 sub
```

经过这样的配置之后，相当于原先在整网拓扑中被孤立的两个不连续子网10.0.12.0/24和10.0.45.0/24网段被新添加的子网10.0.23.0/24和10.0.34.0/24网段连接了起来，即现在已经构成了一个连续的子网。

配置完成后，观察每台路由器的路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 10        Routes : 10
Destination/Mask    Proto   Pre  Cost   Flags  NextHop        Interface
      10.0.0.0/8    RIP     100  1      D      10.0.12.2      Ethernet1/0/0
     10.0.12.0/24   Direct  0    0      D      10.0.12.1      Ethernet1/0/0
     10.0.12.1/32   Direct  0    0      D      127.0.0.1      Ethernet1/0/0
     10.0.23.0/24   RIP     100  1      D      10.0.12.2      Ethernet1/0/0
     10.0.34.0/24   RIP     100  2      D      10.0.12.2      Ethernet1/0/0
     10.0.45.0/24   RIP     100  3      D      10.0.12.2      Ethernet1/0/0
      127.0.0.0/8   Direct  0    0      D      127.0.0.1      InLoopBack0
      127.0.0.1/32  Direct  0    0      D      127.0.0.1      InLoopBack0
  192.168.23.0/24   RIP     100  1      D      10.0.12.2      Ethernet1/0/0
  192.168.34.0/24   RIP     100  2      D      10.0.12.2      Ethernet1/0/0

[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 13        Routes : 15
Destination/Mask    Proto   Pre  Cost   Flags  NextHop        Interface
      10.0.0.0/8    RIP     100  1      D      10.0.23.3      Serial2/0/0
                    RIP     100  1      D      192.168.23.3   Serial2/0/0
     10.0.12.0/24   Direct  0    0      D      10.0.12.2      Ethernet1/0/0
     10.0.12.2/32   Direct  0    0      D      127.0.0.1      Ethernet1/0/0
     10.0.23.0/24   Direct  0    0      D      10.0.23.2      Serial2/0/0
     10.0.23.2/32   Direct  0    0      D      127.0.0.1      Serial2/0/0
     10.0.34.0/24   RIP     100  1      D      10.0.23.3      Serial2/0/0
     10.0.45.0/24   RIP     100  2      D      10.0.23.3      Serial2/0/0
      127.0.0.0/8   Direct  0    0      D      127.0.0.1      InLoopBack0
      127.0.0.1/32  Direct  0    0      D      127.0.0.1      InLoopBack0
  192.168.23.0/24   Direct  0    0      D      192.168.23.2   Serial2/0/0
  192.168.23.2/32   Direct  0    0      D      127.0.0.1      Serial2/0/0
  192.168.23.3/32   Direct  0    0      D      192.168.23.3   Serial2/0/0
  192.168.34.0/24   RIP     100  1      D      192.168.23.3   Serial2/0/0
                    RIP     100  1      D      10.0.23.3      Serial2/0/0
```

```
[R3]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 15        Routes : 16
Destination/Mask    Proto   Pre  Cost   Flags   NextHop         Interface
10.0.0.0/8          RIP     100  1      D       192.168.23.2    Serial1/0/0
                    RIP     100  1      D       192.168.34.4    Serial1/0/1
10.0.12.0/24        RIP     100  1      D       10.0.23.2       Serial1/0/0
10.0.23.0/24        Direct  0    0      D       10.0.23.3       Serial1/0/0
10.0.23.3/32        Direct  0    0      D       127.0.0.1       Serial1/0/0
10.0.34.0/24        Direct  0    0      D       10.0.34.3       Serial1/0/1
10.0.34.3/32        Direct  0    0      D       127.0.0.1       Serial1/0/1
10.0.45.0/24        RIP     100  1      D       10.0.34.4       Serial1/0/1
127.0.0.0/8         Direct  0    0      D       127.0.0.1       InLoopBack0
127.0.0.1/32        Direct  0    0      D       127.0.0.1       InLoopBack0
192.168.23.0/24     Direct  0    0      D       192.168.23.3    Serial1/0/0
192.168.23.2/32     Direct  0    0      D       192.168.23.2    Serial1/0/0
192.168.23.3/32     Direct  0    0      D       127.0.0.1       Serial1/0/0
192.168.34.0/24     Direct  0    0      D       192.168.34.3    Serial1/0/1
192.168.34.3/32     Direct  0    0      D       127.0.0.1       Serial1/0/1
192.168.34.4/32     Direct  0    0      D       192.168.34.4    Serial1/0/1

[R4]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 13        Routes : 15
Destination/Mask    Proto   Pre  Cost   Flags   NextHop         Interface
10.0.0.0/8          RIP     100  1      D       192.168.34.3    Serial2/0/1
                    RIP     100  1      D       10.0.34.3       Serial2/0/1
10.0.12.0/24        RIP     100  2      D       10.0.34.3       Serial2/0/1
10.0.23.0/24        RIP     100  1      D       10.0.34.3       Serial2/0/1
10.0.34.0/24        Direct  0    0      D       10.0.34.4       Serial2/0/1
10.0.34.4/32        Direct  0    0      D       127.0.0.1       Serial2/0/1
10.0.45.0/24        Direct  0    0      D       10.0.45.4       Ethernet1/0/0
10.0.45.4/32        Direct  0    0      D       127.0.0.1       Ethernet1/0/0
127.0.0.0/8         Direct  0    0      D       127.0.0.1       InLoopBack0
127.0.0.1/32        Direct  0    0      D       127.0.0.1       InLoopBack0
192.168.23.0/24     RIP     100  1      D       192.168.34.3    Serial2/0/1
                    RIP     100  1      D       10.0.34.3       Serial2/0/1
192.168.34.0/24     Direct  0    0      D       192.168.34.4    Serial2/0/1
192.168.34.3/32     Direct  0    0      D       192.168.34.3    Serial2/0/1
192.168.34.4/32     Direct  0    0      D       127.0.0.1       Serial2/0/1

[R5]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 10        Routes : 10
Destination/Mask    Proto   Pre  Cost   Flags   NextHop         Interface
  10.0.0.0/8        RIP     100  1      D       10.0.45.4       Ethernet1/0/0
  10.0.12.0/24      RIP     100  3      D       10.0.45.4       Ethernet1/0/0
  10.0.23.0/24      RIP     100  2      D       10.0.45.4       Ethernet1/0/0
  10.0.34.0/24      RIP     100  1      D       10.0.45.4       Ethernet1/0/0
```

```
10.0.45.0/24        Direct   0     0     D     10.0.45.5       Ethernet1/0/0
10.0.45.5/32        Direct   0     0     D     127.0.0.1       Ethernet1/0/0
127.0.0.0/8         Direct   0     0     D     127.0.0.1       InLoopBack0
127.0.0.1/32        Direct   0     0     D     127.0.0.1       InLoopBack0
192.168.23.0/24     RIP      100   2     D     10.0.45.4       Ethernet1/0/0
192.168.34.0/24     RIP      100   1     D     10.0.45.4       Ethernet1/0/0
```

此时发现在每台路由器的路由表中都拥有了所有的子网信息。其中在 R2 上，由于 R3 的 S 0/0/0 接口配置了第二 IP 地址，所以下一跳为 R3 的 S 0/0/0 接口的路由条目会出现两个下一跳地址。

在 R1 上测试与 R5 之间的连通性。

```
[R1]ping 10.0.45.5
  PING 10.0.45.5: 56   data bytes, press CTRL_C to break
    Reply from 10.0.45.5: bytes=56 Sequence=1 ttl=252 time=110 ms
    Reply from 10.0.45.5: bytes=56 Sequence=2 ttl=252 time=140 ms
    Reply from 10.0.45.5: bytes=56 Sequence=3 ttl=252 time=80 ms
    Reply from 10.0.45.5: bytes=56 Sequence=4 ttl=252 time=100 ms
    Reply from 10.0.45.5: bytes=56 Sequence=5 ttl=252 time=90 ms
  --- 10.0.45.5 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 80/104/140 ms
```

连通性测试成功。

上述做法通过在不连续的子网之间的链路上配置相同主网的子网 IP 地址，即采用配置从 IP 地址的方式来实现子网的连续性，解决了因为自动汇总发生后，子网路由被抑制掉而导致的子网不可达。此种做法优点是 RIPv1 在不做大的拓扑结构调整的前提下，仅靠配置第二个 IP 地址就解决了不连续子网问题；不足之处是需要配置第二个 IP 地址，要消耗掉多个子网网段。

4. RIPv2 中解决不连续子网问题

如果路由器运行的是 RIPv2，则可以直接关闭自动汇总，子网是否连续就不重要了，因为 RIPv2 会直接通告相应的子网路由。

删除上一步骤中的从 IP 地址配置命令，并在所有路由器中将 RIP 的版本配置为 2，且关闭自动汇总。

```
[R1]rip
[R1-rip-1]version 2
[R1-rip-1]undo summary

[R2]interface serial 2/0/0
[R2-Serial2/0/0]undo ip address 10.0.23.2 24 sub
[R2-Serial2/0/0]rip
[R2-rip-1]version 2
[R2-rip-1]undo summary

[R3]interface serial 1/0/0
[R3-Serial1/0/0]undo ip address 10.0.23.3 24 sub
[R3-Serial1/0/0]interface serial 1/0/1
[R3-Serial1/0/1]undo ip address 10.0.34.3 24 sub
[R3-Serial1/0/1]rip
[R3-rip-1]version 2
```

```
[R3-rip-1]undo summary

[R4]interface serial 2/0/1
[R4-Serial2/0/1]undo ip address 10.0.34.4 24 sub
[R4-Serial2/0/1]rip
[R4-rip-1]version 2
[R4-rip-1]undo summary

[R5]rip
[R5-rip-1]version 2
[R5-rip-1]undo summary
```

配置完成后，观察每台路由器上的路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7        Routes : 7
Destination/Mask    Proto   Pre  Cost     Flags  NextHop         Interface
      10.0.12.0/24  Direct  0    0        D      10.0.12.1       Ethernet1/0/0
      10.0.12.1/32  Direct  0    0        D      127.0.0.1       Ethernet1/0/0
      10.0.45.0/24  RIP     100  3        D      10.0.12.2       Ethernet1/0/0
       127.0.0.0/8  Direct  0    0        D      127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0        D      127.0.0.1       InLoopBack0
   192.168.23.0/24  RIP     100  1        D      10.0.12.2       Ethernet1/0/0
   192.168.34.0/24  RIP     100  2        D      10.0.12.2       Ethernet1/0/0

[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 9        Routes : 9
Destination/Mask    Proto   Pre  Cost     Flags  NextHop         Interface
      10.0.12.0/24  Direct  0    0        D      10.0.12.2       Ethernet1/0/0
      10.0.12.2/32  Direct  0    0        D      127.0.0.1       Ethernet1/0/0
      10.0.45.0/24  RIP     100  2        D      192.168.23.3    Serial2/0/0
       127.0.0.0/8  Direct  0    0        D      127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0        D      127.0.0.1       InLoopBack0
   192.168.23.0/24  Direct  0    0        D      192.168.23.2    Serial2/0/0
   192.168.23.2/32  Direct  0    0        D      127.0.0.1       Serial2/0/0
   192.168.23.3/32  Direct  0    0        D      192.168.23.3    Serial2/0/0
   192.168.34.0/24  RIP     100  1        D      192.168.23.3    Serial2/0/0

[R3]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 10       Routes : 10
Destination/Mask    Proto   Pre  Cost     Flags  NextHop         Interface
      10.0.12.0/24  RIP     100  1        D      192.168.23.2    Serial1/0/0
      10.0.45.0/24  RIP     100  1        D      192.168.34.4    Serial1/0/1
       127.0.0.0/8  Direct  0    0        D      127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0        D      127.0.0.1       InLoopBack0
   192.168.23.0/24  Direct  0    0        D      192.168.23.3    Serial1/0/0
   192.168.23.2/32  Direct  0    0        D      192.168.23.2    Serial1/0/0
```

```
192.168.23.3/32  Direct  0    0    D   127.0.0.1      Serial1/0/0
192.168.34.0/24  Direct  0    0    D   192.168.34.3   Serial1/0/1
192.168.34.3/32  Direct  0    0    D   127.0.0.1      Serial1/0/1
192.168.34.4/32  Direct  0    0    D   192.168.34.4   Serial1/0/1

[R4]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 9        Routes : 9
Destination/Mask    Proto   Pre  Cost   Flags  NextHop         Interface
      10.0.12.0/24  RIP     100  2      D      192.168.34.3    Serial2/0/1
      10.0.45.0/24  Direct  0    0      D      10.0.45.4       Ethernet1/0/0
      10.0.45.4/32  Direct  0    0      D      127.0.0.1       Ethernet1/0/0
      127.0.0.0/8   Direct  0    0      D      127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0      D      127.0.0.1       InLoopBack0
   192.168.23.0/24  RIP     100  1      D      192.168.34.3    Serial2/0/1
   192.168.34.0/24  Direct  0    0      D      192.168.34.4    Serial2/0/1
   192.168.34.3/32  Direct  0    0      D      192.168.34.3    Serial2/0/1
   192.168.34.4/32  Direct  0    0      D      127.0.0.1       Serial2/0/1

[R5]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7        Routes : 7
Destination/Mask    Proto   Pre  Cost   Flags  NextHop         Interface
      10.0.12.0/24  RIP     100  3      D      10.0.45.4       Ethernet1/0/0
      10.0.45.0/24  Direct  0    0      D      10.0.45.5       Ethernet1/0/0
      10.0.45.5/32  Direct  0    0      D      127.0.0.1       Ethernet1/0/0
      127.0.0.0/8   Direct  0    0      D      127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0      D      127.0.0.1       InLoopBack0
   192.168.23.0/24  RIP     100  2      D      10.0.45.4       Ethernet1/0/0
   192.168.34.0/24  RIP     100  1      D      10.0.45.4       Ethernet1/0/0
```

可以看到在所有路由表中都没有汇总路由 10.0.0.0/8，并且 10.0.45.0/24 和 10.0.12.0/24 子网出现在所有的路由表中。

测试 R1、R5 间的连通性。

```
[R1]ping 10.0.45.5
  PING 10.0.45.5: 56  data bytes, press CTRL_C to break
    Reply from 10.0.45.5: bytes=56 Sequence=1 ttl=252 time=90 ms
    Reply from 10.0.45.5: bytes=56 Sequence=2 ttl=252 time=130 ms
    Reply from 10.0.45.5: bytes=56 Sequence=3 ttl=252 time=90 ms
    Reply from 10.0.45.5: bytes=56 Sequence=4 ttl=252 time=110 ms
    Reply from 10.0.45.5: bytes=56 Sequence=5 ttl=252 time=90 ms
  --- 10.0.45.5 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 90/102/130 ms
```

在 RIPv2 的环境中，因为默认情况下自动汇总是开启的，所以在设计网络时，应尽量不要出现同主网的子网被其他主网分隔的情况。如果出现了，关闭自动汇总是最佳的做法，不足之处是路由表中路由条目会增加。

思考

在使用 RIPv1 的环境中，R2、R3 和 R4 都配置了第二个 IP 地址，10.0.0.0/8 的子网已经连续，如果 R2 是主网边界，为什么 R3 还能看到 10.0.12.0/24 的子网？如果 R2 不是主网边界，为什么在 R3 的路由表里能看到 10.0.0.0/8 的汇总路由？

7.7 RIP 的水平分割及触发更新

原理概述

水平分割（Split Horizon）指的是 RIP 从某个接口接收到的路由信息，不会从该接口再发回给邻居设备。这样不但减少了带宽消耗，还可以防止路由环路。在华为设备上，水平分割功能默认情况下是开启的。

触发更新（Triggered Updates）的原理是，当路由信息发生变化时，运行 RIP 的设备会立即向邻居设备发送更新报文，而不必等待定时更新，从而缩短了网络收敛时间。在华为设备上，没有相关命令能主动关闭触发更新的功能。

毒性逆转（Poison Reverse）指的是 RIP 从某个接口接收到路由信息后，将该路由的开销设置为 16（即该路由不可达），并从原接口发回邻居设备。利用这种方式，可以清除对方路由表中的无用路由。如果同时都配置了毒性逆转和水平分割，水平分割行为会被毒性逆转行为代替。在华为设备上，毒性逆转功能默认情况下是关闭的，需要手动打开此功能。

毒性逆转可以快速清除无用的路由而不必等待老化时间，另外，在帧中继和 X.25 等非广播多路访问网络中，如果开启了水平分割功能，会造成有的路由器无法接收到更新路由的情况，因此在这种网络中，水平分割是默认禁止的，我们需要手动开启毒性逆转防止路由环路。

RIPv1 和 RIPv2 都支持水平分割、触发更新和毒性逆转功能。

实验目的

- 掌握 RIP 触发更新的原理
- 掌握 RIP 中触发更新与等待老化时间的现象差别
- 掌握 RIP 水平分割的原理和配置
- 掌握开启水平分割与关掉水平分割时路由条目的变化
- 掌握毒性逆转原理和配置

实验内容

本实验模拟企业网络场景。R1 为该公司出口网关路由器，连接运营商网络，R2 为公司 IT 部门路由器，通过交换机 S2 与网关相连；同时公司人力资源部路由器 R3 也通过交换机 S1 与网关相连，所有路由器运行路由协议 RIPv2 实现公司整网互通。当 R3 与

S1 之间链路 down 掉时 R1 不会触发更新，网络收敛较慢；而当 R1 与 S1 之间链路 down 掉时，R1 会触发更新，网络收敛快。华为路由器默认开启 RIP 水平分割功能，当主动关掉路由器上水平分割功能时，检查路由器发送 RIP 路由条目的变化。手动开启毒性逆转功能，检查路由器发送 RIP 路由条目的变化。

实验拓扑

RIP 的水平分割及触发更新的拓扑如图 7-16 所示。

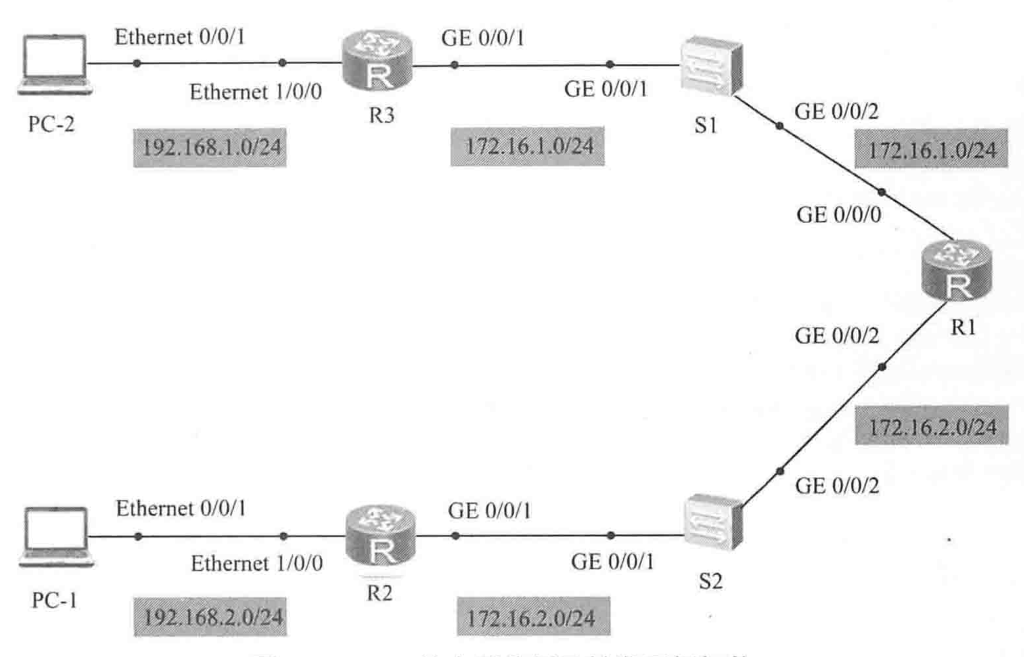


图 7-16 RIP 的水平分割及触发更新拓扑

实验编址

实验编址见表 7-7。

表 7-7 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR2220）	GE 0/0/0	172.16.1.1	255.255.255.0	N/A
	GE 0/0/2	172.16.2.1	255.255.255.0	N/A
R2（AR2220）	GE 0/0/1	172.16.2.2	255.255.255.0	N/A
	Ethernet 1/0/0	192.168.2.254	255.255.255.0	N/A
R3（AR2220）	GE 0/0/1	172.16.1.2	255.255.255.0	N/A
	Ethernet 1/0/0	192.168.1.254	255.255.255.0	N/A
PC-1	Ethernet 0/0/1	192.168.2.1	255.255.255.0	192.168.2.254
PC-2	Ethernet 0/0/1	192.168.1.1	255.255.255.0	192.168.1.254

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 地址配置，并使用 **ping** 命令检测各直连链路的

连通性。

```
[R1]ping -c 1 172.16.1.2
  PING 172.16.1.2: 56  data bytes, press CTRL_C to break
    Reply from 172.16.1.2: bytes=56 Sequence=1 ttl=255 time=80 ms
  --- 172.16.1.2 ping statistics ---
    1 packet(s) transmitted
    1 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 80/80/80 ms
```

其余直连网段的连通性测试省略。

2. 搭建 RIP 网络

根据图 7-16，在 R1、R2、R3 上配置 RIP 协议，并将相应网段通告进 RIP 协议中。

```
[R1]rip 1
[R1-rip-1]version 2
[R1-rip-1]network 172.16.0.0

[R2]rip 1
[R2-rip-1]version 2
[R2-rip-1]network 192.168.2.0
[R2-rip-1]network 172.16.0.0

[R3]rip 1
[R3-rip-1]version 2
[R3-rip-1]network 172.16.0.0
[R3-rip-1]network 192.168.1.0
```

配置完成后，查看 R1 的路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8        Routes : 8
Destination/Mask    Proto    Pre   Cost    Flags    NextHop       Interface
      127.0.0.0/8   Direct   0     0       D        127.0.0.1     InLoopBack0
      127.0.0.1/32  Direct   0     0       D        127.0.0.1     InLoopBack0
     172.16.1.0/24  Direct   0     0       D        172.16.1.1    GigabitEthernet0/0/0
     172.16.1.1/32  Direct   0     0       D        127.0.0.1     GigabitEthernet0/0/0
     172.16.2.0/24  Direct   0     0       D        172.16.2.1    GigabitEthernet0/0/2
     172.16.2.1/32  Direct   0     0       D        127.0.0.1     GigabitEthernet0/0/2
192.168.1.0/24      RIP      100   1       D        172.16.1.2    GigabitEthernet0/0/0
192.168.2.0/24      RIP      100   1       D        172.16.2.2    GigabitEthernet0/0/2
```

可以观察到，R1 已经正常获取到了 PC-1 与 PC-2 所在网段的 RIP 路由信息。R2、R3 上的查看过程省略。

3. 验证触发更新

断掉 R3 与 S1 之间的链路（在模拟器上直接删除该链路），然后查看 R2 的路由表。

```
[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8        Routes : 8
Destination/Mask    Proto    Pre   Cost    Flags    NextHop       Interface
  127.0.0.0/8       Direct   0     0       D        127.0.0.1     InLoopBack0
```

```
  127.0.0.1/32        Direct   0    0    D    127.0.0.1        InLoopBack0
  172.16.1.0/24       RIP      100  1    D    172.16.2.1       GigabitEthernet0/0/1
  172.16.2.0/24       Direct   0    0    D    172.16.2.2       GigabitEthernet0/0/1
  172.16.2.2/32       Direct   0    0    D    127.0.0.1        GigabitEthernet0/0/1
192.168.1.0/24        RIP      100  2    D    172.16.2.1       GigabitEthernet0/0/1
192.168.2.0/24        Direct   0    0    D    192.168.2.254    Ethernet1/0/0
  192.168.2.254/32    Direct   0    0    D    127.0.0.1        Ethernet1/0/0
```

可以观察到 192.168.1.0 网段的路由信息仍然存在。这是因为断掉的不是 R1 的直连接口，R1 此时无法直接感知到故障的发生，路由条目需要等 180s 的老化计时器超时后，此路由条目才会在路由表中删除。

等待 180s 老化时间后，查看 R2 的路由表。

```
[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8          Routes : 8
Destination/Mask      Proto    Pre  Cost  Flags  NextHop          Interface
  127.0.0.0/8         Direct   0    0     D      127.0.0.1        InLoopBack0
  127.0.0.1/32        Direct   0    0     D      127.0.0.1        InLoopBack0
  172.16.1.0/24       RIP      100  1     D      172.16.2.1       GigabitEthernet0/0/1
  172.16.2.0/24       Direct   0    0     D      172.16.2.2       GigabitEthernet0/0/1
  172.16.2.2/32       Direct   0    0     D      127.0.0.1        GigabitEthernet0/0/1
  192.168.2.0/24      Direct   0    0     D      192.168.2.254    Ethernet1/0/0
  192.168.2.254/32    Direct   0    0     D      127.0.0.1        Ethernet1/0/0
```

可以观察到此时 192.168.1.0 网段的路由信息已经从路由表中删除。

恢复 R3 与 S1 之间的链路（在模拟器上重新连线），并在 R2 的路由表正常后，断掉 R1 与 S1 之间的链路，查看 R2 路由表。

```
[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 6          Routes : 6
Destination/Mask      Proto    Pre  Cost  Flags  NextHop          Interface
  127.0.0.0/8         Direct   0    0     D      127.0.0.1        InLoopBack0
  127.0.0.1/32        Direct   0    0     D      127.0.0.1        InLoopBack0
  172.16.2.0/24       Direct   0    0     D      172.16.2.2       GigabitEthernet0/0/1
  172.16.2.2/32       Direct   0    0     D      127.0.0.1        GigabitEthernet0/0/1
192.168.2.0/24        Direct   0    0     D      192.168.2.254    Ethernet1/0/0
192.168.2.254/32      Direct0  0          D      127.0.0.1        Ethernet1/0/0
```

可以观察到 192.168.1.0 网段的路由信息已经不存在，因为断掉的是 R1 的直连接口，所以 R1 能够直接感知到链路发生故障，在路由表删除 192.168.1.0 的路由同时，并会触发更新，使得 R2 上的路由表为最新状态。

4. 验证水平分割

在 R2 上使用 **debugging rip 1 send GigabitEthernet 0/0/1** 命令打开 debug 功能，再用 **terminal monitor**、**terminal debugging** 命令查看 R2 发送给 R1 的路由条目。

```
<R2>debugging rip 1 send GigabitEthernet 0/0/1
<R2>terminal monitor
<R2>terminal debugging
<R2>
```

```
Jun 19 2013 19:17:59.640.1-08:00 R2 RIP/7/DBG: 6: 12227: RIP 1: Sending response
on interface GigabitEthernet0/0/1 from 172.16.2.2 to 224.0.0.9
Jun 19 2013 19:17:59.640.2-08:00 R2 RIP/7/DBG: 6: 12247: Packet: Version 2, Cmd response, Length 24
Jun 19 2013 19:17:59.640.3-08:00 R2 RIP/7/DBG: 6: 12315: Dest 192.168.2.0/24, Nexthop 0.0.0.0, Cost 1, Tag 0
```

从 debug 的信息中可以观察到 R2 发送给 R1 的路由条目中没有包含 192.168.1.0 网段的路由信息，因为该路由条目是从 R1 始发过来的。

下面关闭 debug，并在 R2 的 GE 0/0/1 和 R1 的 GE 0/0/2 接口下使用 **undo rip split-horizon** 命令关闭水平分割功能。

```
<R2>undo debugging all
<R2>system-view
[R2]interface GigabitEthernet0/0/1
[R2-GigabitEthernet0/0/1]undo rip split-horizon

[R1]interface GigabitEthernet0/0/2
[R1-GigabitEthernet0/0/2]undo rip split-horizon
```

配置完成后查看 debug 信息。

```
<R2>debugging rip 1 send GigabitEthernet 0/0/1
<R2>terminal monitor
<R2>terminal debugging
Jun 19 2013 19:21:53.910.4-08:00 R2 RIP/7/DBG: 6: 12315: Dest 172.16.0.0/16, Next
hop 0.0.0.0, Cost 1, Tag 0
Jun 19 2013 19:21:53.910.5-08:00 R2 RIP/7/DBG: 6: 12315: Dest 192.168.1.0/24, Nexthop 0.0.0.0, Cost 3, Tag 0
Jun 19 2013 19:21:53.910.6-08:00 R2 RIP/7/DBG: 6: 12315: Dest 192.168.2.0/24, Nexthop 0.0.0.0, Cost 1, Tag 0
```

从 debug 信息中可以观察到 R2 发送给 R1 的路由条目中包含有 192.168.1.0 网段，此时接口上的水平分割功能不生效。

5. 验证毒性逆转

关闭 debug，并在 R2 的 GE 0/0/1 接口下恢复水平分割功能。

```
<R2>undo debugging all
<R2>system-view
[R2]interface GigabitEthernet0/0/1
[R2-GigabitEthernet0/0/1]rip split-horizon
```

配置完成后，在 R2 上开启 debug 功能。

```
<R2>debugging rip 1 send GigabitEthernet 0/0/1
<R2>terminal monitor
<R2>terminal debugging
<R2>
Jun 19 2013 19:28:06.720.1-08:00 R2 RIP/7/DBG: 6: 12227: RIP 1: Sending response
on interface GigabitEthernet0/0/1 from 172.16.2.2 to 224.0.0.9
Jun 19 2013 19:28:06.720.2-08:00 R2 RIP/7/DBG: 6: 12247: Packet: Version 2, Cmd response, Length 24
Jun 19 2013 19:28:06.720.3-08:00 R2 RIP/7/DBG: 6: 12315: Dest 192.168.2.0/24, Nexthop 0.0.0.0, Cost 1, Tag 0
```

通过 debug 信息可以观察到，此时开启了水平分割后，R2 发送给 R1 的路由条目中没有包含 192.168.1.0 网段。

关闭 debug，并在 R2 的 GE 0/0/1 接口下使用 **rip poison-reverse** 命令开启毒性逆转功能。

```
<R2>undo debugging all
<R2>system-view
[R2]interface GigabitEthernet0/0/1
[R2-GigabitEthernet0/0/1]rip poison-reverse
```

配置完成后，查看 debug 信息。

```
<R2>debugging rip 1 send GigabitEthernet 0/0/1
<R2>terminal monitor
<R2>terminal debugging
Jun 19 2013 19:30:00.520.2-08:00 R2 RIP/7/DBG: 6: 12247: Packet: Version 2, Cmd r • esponse, Length 64
Jun 19 2013 19:30:00.520.4-08:00 R2 RIP/7/DBG: 6: 12315: Dest 192.168.1.0/24, Nexthop 172.16.2.1, Cost 16, Tag 0
Jun 19 2013 19:30:00.520.5-08:00 R2 RIP/7/DBG: 6: 12315: Dest 192.168.2.0/24, Nexthop 0.0.0.0, Cost 1, Tag 0
```

可以观察到，R2 发送给 R1 的路由条目中包含了 192.168.1.0 网段，但是 cost 值为 16。说明在毒性逆转和水平分割同时开启的情况下，简单的水平分割行为（从某接口学到的路由再从这个接口发布时将被抑制）会被毒性逆转行为代替。

思考

水平分割可以防止环路，那为什么 RIP 协议还需要其他防环机制？水平分割的局限性在哪儿？

7.8　配置 RIP 路由附加度量值

原理概述

路由附加度量值是在 RIP 路由原来度量值的基础上所增加或减少的度量值（跳数）。对于 RIP 接收和发布路由，可通过不同的命令配置附加度量值更加灵活地控制 RIP 的路由选择。

rip metricin 命令用于在接收到路由后，给其增加一个附加度量值，再加入路由表中，使得路由表中的度量值发生变化。运行该命令会影响到本地设备和其他设备的路由选择。

rip metricout 命令用于自身路由的发布，发布时增加一个附加的度量值，但本地路由表中的度量值不会发生变化。运行该命令不会影响本地设备的路由选择，但是会影响其他设备的路由选择。

实验目的

- 理解 RIP 路由附加度量值应用场景
- 掌握使用 metricin 方式附加度量值的方法
- 掌握使用 metricout 方式附加度量值的方法

实验内容

本实验模拟公司网络场景。路由器 R1 左侧连接的是公司市场部，路由器 R4 右侧连接的是公司财务部，R1 与 R4 之间通过 R2、R3 双链路连接，所有路由器运行 RIP 协议，R1 与 R4 之间互访的流量通过两条链路负载分担。现在网络管理员在 R2 上做了流量控制，要求所有市场部访问财务部的流量都必须经过 R2，同时为了减轻 R2 的负担，由财务部去往市场部的流量都由 R3 来转发。网络管理员可通过两种 RIP 路由附加度量值的方式修改相应的路由度量值，灵活控制 RIP 的路由选择来达到公司的流量控制。

实验拓扑

配置 RIP 路由附加度量值的拓扑如图 7-17 所示。

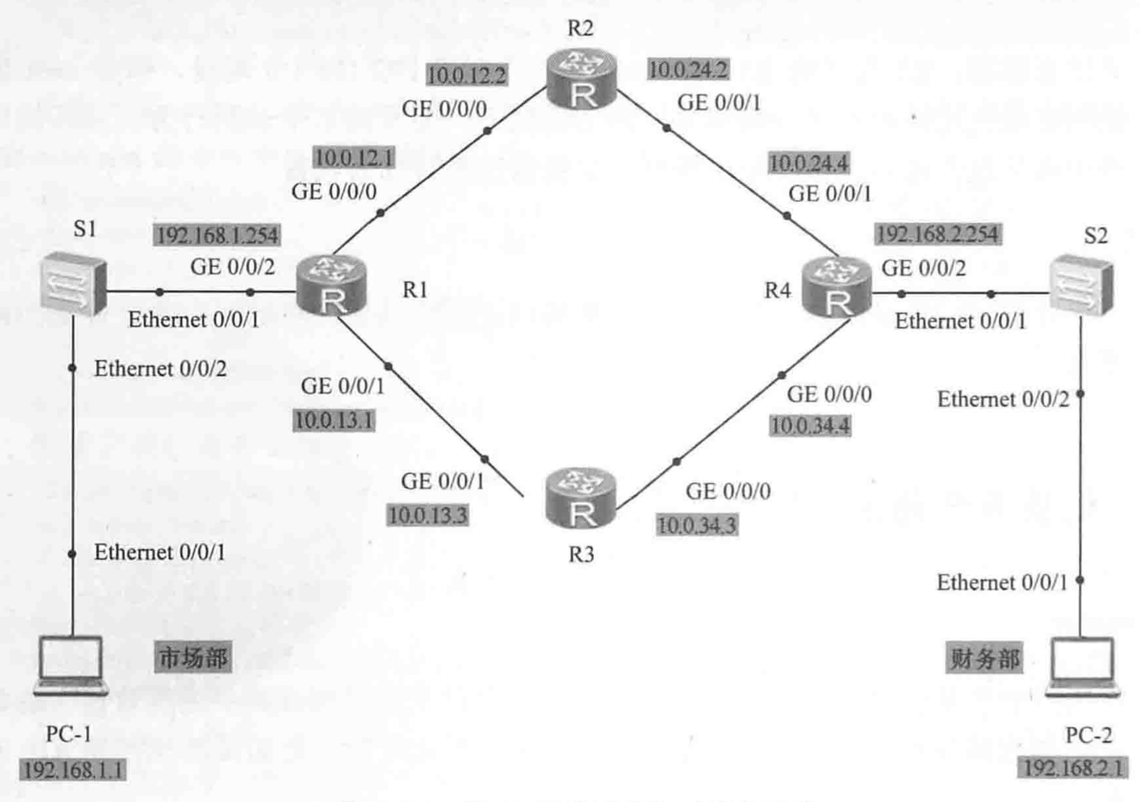


图 7-17 配置 RIP 路由附加度量值拓扑

实验编址

实验编址见表 7-8。

表 7-8 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR2220）	GE 0/0/0	10.0.12.1	255.255.255.0	N/A
	GE 0/0/1	10.0.13.1	255.255.255.0	N/A
	GE 0/0/2	192.168.1.254	255.255.255.0	N/A
R2（AR2220）	GE 0/0/0	10.0.12.2	255.255.255.0	N/A
	GE 0/0/1	10.0.24.2	255.255.255.0	N/A
R3（AR2220）	GE 0/0/0	10.0.34.3	255.255.255.0	N/A
	GE 0/0/1	10.0.13.3	255.255.255.0	N/A
R4（AR2220）	GE 0/0/0	10.0.34.4	255.255.255.0	N/A
	GE 0/0/1	10.0.24.4	255.255.255.0	N/A
	GE 0/0/2	192.168.2.254	255.255.255.0	N/A
PC-1	Ethernet 0/0/1	192.168.1.1	255.255.255.0	192.168.1.254
PC-2	Ethernet 0/0/1	192.168.2.1	255.255.255.0	192.168.2.254

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 地址配置，并使用 **ping** 命令检测各直连链路的连通性。

```
[R1]ping 10.0.12.2
  PING 10.0.12.2: 56   data bytes, press CTRL_C to break
    Reply from 10.0.12.2: bytes=56 Sequence=1 ttl=255 time=50 ms
    Reply from 10.0.12.2: bytes=56 Sequence=2 ttl=255 time=10 ms
    Reply from 10.0.12.2: bytes=56 Sequence=3 ttl=255 time=10 ms
    Reply from 10.0.12.2: bytes=56 Sequence=4 ttl=255 time=10 ms
    Reply from 10.0.12.2: bytes=56 Sequence=5 ttl=255 time=1 ms
  --- 10.0.12.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 1/16/50 ms
```

其他直连网段的连通性检测省略。

2. 搭建 RIP 网络

公司内部网络使用 RIP 协议。首先配置 R1、R2、R3 和 R4 运行 RIP 协议，通告所有网段，使公司网络互通。

```
[R1]rip 1
[R1-rip-1]version 2
[R1-rip-1]undo summary
[R1-rip-1]network 10.0.0.0
[R1-rip-1]network 192.168.1.0
```

其他路由器配置省略。配置完成后查看 R1 的路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 16        Routes : 17
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      10.0.12.0/24  Direct  0    0           D   10.0.12.1       GigabitEthernet0/0/0
      10.0.12.1/32  Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/0
    10.0.12.255/32  Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/0
      10.0.13.0/24  Direct  0    0           D   10.0.13.1       GigabitEthernet0/0/1
      10.0.13.1/32  Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/1
    10.0.13.255/32  Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/1
      10.0.24.0/24  RIP     100  1           D   10.0.12.2       GigabitEthernet0/0/0
      10.0.34.0/24  RIP     100  1           D   10.0.13.3       GigabitEthernet0/0/1
       127.0.0.0/8  Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
127.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0
    192.168.1.0/24  Direct  0    0           D   192.168.1.254   GigabitEthernet0/0/2
  192.168.1.254/32  Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/2
  192.168.1.255/32  Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/2
    192.168.2.0/24  RIP     100  2           D   10.0.12.2       GigabitEthernet0/0/0
                    RIP     100  2           D   10.0.13.3       GigabitEthernet0/0/1
255.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0
```

可以观察到在 R1 上存在两条通过 RIP 协议接收到的去往财务部所在网段 192.168.2.0/24 的路由条目。

同样在 R4 上查看路由表。

```
<R4>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 16        Routes : 17
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      10.0.12.0/24  RIP     100  1           D   10.0.24.2       GigabitEthernet0/0/1
      10.0.13.0/24  RIP     100  1           D   10.0.34.3       GigabitEthernet0/0/0
      10.0.24.0/24  Direct  0    0           D   10.0.24.4       GigabitEthernet0/0/1
      10.0.24.4/32  Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/1
    10.0.24.255/32  Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/1
      10.0.34.0/24  Direct  0    0           D   10.0.34.4       GigabitEthernet0/0/0
      10.0.34.4/32  Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/0
    10.0.34.255/32  Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/0
       127.0.0.0/8  Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
127.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0
    192.168.1.0/24  RIP     100  2           D   10.0.34.3       GigabitEthernet0/0/0
                    RIP     100  2           D   10.0.24.2       GigabitEthernet0/0/1
    192.168.2.0/24  Direct  0    0           D   192.168.2.254   GigabitEthernet0/0/2
  192.168.2.254/32  Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/2
  192.168.2.255/32  Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/2
255.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0
```

同样可以观察到在 R4 上存在两条通过 RIP 协议接收到的去往市场部所在网段 192.168.1.0/24 的路由条目，且有两个下一跳，呈现负载分担。

3. 配置RIP Metricin

在 R1 的 GE 0/0/1 接口下使用**rip metricin 2 命令**，设置 R1 在接收 R3 发送来的路由条目时增加度量值 2。这样由 R3 发给 R1 的路由条目的度量值将大于 R2 发给 R1 的路由，R1 会优选 R2 发来的 RIP 路由条目，并加入路由表中。

```
[R1]interface GigabitEthernet 0/0/1
[R1-GigabitEthernet0/0/1]rip metricin 2
```

配置完成后，查看路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 16        Routes : 16
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      10.0.12.0/24  Direct  0    0           D   10.0.12.1       GigabitEthernet0/0/0
      10.0.12.1/32  Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/0
    10.0.12.255/32  Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/0
      10.0.13.0/24  Direct  0    0           D   10.0.13.1       GigabitEthernet0/0/1
      10.0.13.1/32  Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/1
    10.0.13.255/32  Direct  0    0           D   127.0.0.1       GigabitEthernet0/0/1
      10.0.24.0/24  RIP     100  1           D   10.0.12.2       GigabitEthernet0/0/0
      10.0.34.0/24  RIP     100  3           D   10.0.13.3       GigabitEthernet0/0/1
       127.0.0.0/8  Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
```

```
127.255.255.255/32  Direct  0    0    D  127.0.0.1      InLoopBack0
    192.168.1.0/24  Direct  0    0    D  192.168.1.254  GigabitEthernet0/0/2
  192.168.1.254/32  Direct  0    0    D  127.0.0.1      GigabitEthernet0/0/2
  192.168.1.255/32  Direct  0    0    D  127.0.0.1      GigabitEthernet0/0/2
    192.168.2.0/24  RIP     100  2    D  10.0.12.2      GigabitEthernet0/0/0
255.255.255.255/32  Direct  0    0    D  127.0.0.1      InLoopBack0
```

可以观察到，此时在 R1 上访问财务部网段只有一个下一跳 R2，在 R1 上查看 RIP 的数据库。

```
[R1]display rip 1 database
---------------------------------------------------------------------------
 Advertisement State : [A] - Advertised
                       [I] - Not Advertised/Withdraw
---------------------------------------------------------------------------
   10.0.0.0/8, cost 0, ClassfulSumm
       10.0.12.0/24, cost 0, [A], Rip-interface
       10.0.13.0/24, cost 0, [A], Rip-interface
       10.0.24.0/24, cost 1, [A], nexthop 10.0.12.2
       10.0.34.0/24, cost 16, [I], nexthop 10.0.13.3
       10.0.34.0/24, cost 2, [A], nexthop 10.0.12.2
     192.168.1.0/24, cost 0, ClassfulSumm
     192.168.1.0/24, cost 0, [A], Rip-interface
     192.168.2.0/24, cost 2, ClassfulSumm
     192.168.2.0/24, cost 2, [A], nexthop 10.0.12.2
     192.168.2.0/24, cost 16, [I], nexthop 10.0.13.3
```

可以观察到在 RIP 的数据库中，还是存在下一跳为 R3 的去往 192.168.2.0/24 网段的路由，但 cost 值被设为 16，即不可达。

在 PC-1 上测试访问 PC-2 所经过的网关设备。

```
PC>tracert 192.168.2.1
traceroute to 192.168.2.1, 8 hops max
(ICMP), press Ctrl+C to stop
 1  192.168.1.254  31 ms  31 ms  16 ms
 2  10.0.12.2      31 ms  31 ms  47 ms
 3  10.0.24.4      31 ms  31 ms  16 ms
 4  192.168.2.1    47 ms  47 ms  62 ms
```

可以观察到，数据包此时是经过 R2 转发至 PC-2 的。

4. 配置 RIP Metricout

为了减轻 R2 的负担，所有由财务部去往市场部的流量都由 R3 来转发。

在 R2 上的 GE 0/0/1 接口下使用 **rip metricout 2** 命令，设置 R2 在向 R4 发送路由条目时增加度量值 2。这样 R4 收到来自 R2 的路由的度量值就会大于来自 R3 的路由，R4 会优选来自 R3 的 RIP 路由条目，并加入到路由表中。

```
[R2]interface GigabitEthernet0/0/1
[R2-GigabitEthernet0/0/1]rip metricout 2
```

配置完成后，查看路由表。

```
<R4>display ip routing-table
Route Flags: R - relay, D - download to fib
---------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 16        Routes : 16
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      10.0.12.0/24  RIP     100  2           D   10.0.24.2       GigabitEthernet0/0/1
```

```
       10.0.13.0/24  RIP     100  1    D  10.0.34.3      GigabitEthernet0/0/0
       10.0.24.0/24  Direct  0    0    D  10.0.24.4      GigabitEthernet0/0/1
       10.0.24.4/32  Direct  0    0    D  127.0.0.1      GigabitEthernet0/0/1
     10.0.24.255/32  Direct  0    0    D  127.0.0.1      GigabitEthernet0/0/1
       10.0.34.0/24  Direct  0    0    D  10.0.34.4      GigabitEthernet0/0/0
       10.0.34.4/32  Direct  0    0    D  127.0.0.1      GigabitEthernet0/0/0
     10.0.34.255/32  Direct  0    0    D  127.0.0.1      GigabitEthernet0/0/0
        127.0.0.0/8  Direct  0    0    D  127.0.0.1      InLoopBack0
       127.0.0.1/32  Direct  0    0    D  127.0.0.1      InLoopBack0
 127.255.255.255/32  Direct  0    0    D  127.0.0.1      InLoopBack0
     192.168.1.0/24  RIP     100  2    D  10.0.34.3      GigabitEthernet0/0/0
     192.168.2.0/24  Direct  0    0    D  192.168.2.254  GigabitEthernet0/0/2
   192.168.2.254/32  Direct  0    0    D  127.0.0.1      GigabitEthernet0/0/2
   192.168.2.255/32  Direct  0    0    D  127.0.0.1      GigabitEthernet0/0/2
 255.255.255.255/32  Direct  0    0    D  127.0.0.1      InLoopBack0
```

可以观察到，此时在 R4 上访问市场部网段只有一个下一跳 R3，在 PC-2 上测试访问 PC-1 所经过的网关设备。

```
PC>tracert 192.168.1.1
traceroute to 192.168.1.1, 8 hops max
(ICMP), press Ctrl+C to stop
 1  192.168.2.254   31 ms   31 ms   16 ms
 2  10.0.34.3       31 ms   47 ms   15 ms
 3  10.0.13.1       31 ms   16 ms   47 ms
 4  192.168.1.1     62 ms   47 ms   31 ms
```

可以观察到，数据包此时是经过 R3 转发至 PC-1 的。

思考

无论是配置 **metricin** 还是 **metricout** 都会将所有 RIP 路由条目的度量值增加，如何在完成对财务部路由附加度量值配置的同时不影响其他 RIP 路由的 Metric 值？

7.9 RIP 的故障处理

原理概述

在实际的小型网络中，设计者可能会选用 RIP 作为网络路由协议，但在实际操作配置中，往往会出现人为的疏忽、错误配置等种种问题造成网络不能正常通信。这便需要网络管理员具有一定的故障处理能力，去排除网络中的故障以恢复网络通信。

下面为 RIP 中常见的故障：

（1）接口状态不是 UP；

（2）RIP 进程下没有对该网段做 network 配置；

（3）对端 RIP 协议报文的版本号和本地接收的 RIP 协议报文版本号不一致；

（4）接口上配置了禁止接收 RIP 报文或禁止发送 RIP 报文的命令；

（5）在 RIP 中配置了策略，过滤掉收到的 RIP 路由或不允许发送 RIP 路由；

（6）接口上没有开启水平分割；
（7）链路两端的接口认证方式不匹配；
（8）路由表中存在从其他协议获得的相同路由条目；
（9）收到的路由度量值大于 16。

可以根据以上的常见问题，制定出合理的故障排除步骤。比如检查第一项时，首先使用相关的配置命令查看接口状态是否 UP，这就要求熟悉用于检查各个状态的配置命令。若第一项没有问题，则按顺序查看下一项，直到排查出错误，恢复网络的正常。

实验目的

- 掌握 RIP 故障的常见原因
- 掌握 RIP 故障诊断流程
- 掌握 RIP 故障处理步骤
- 掌握 RIP 故障排除的常用命令

实验内容

本实验模拟企业网络场景。R1 为该公司出口网关路由器，连接运营商网络；R2 为公司 HR 部门路由器与网关相连；由于公司的网络规模比较小，所以选择使用 RIPv2 来作为动态路由协议实现公司整网互通。现在公司 IT 部门员工发现用 PC-2 无法与 HR 部门的 PC-1 通信，作为公司的网络管理员，现需对此网络故障进行排查，恢复网络。两台 PC 都确认了 IP 地址和网关地址设置正确，现给出公司网络拓扑以及 3 台路由器的配置，请用模拟器搭建网络并把已经给出的配置拷贝入对应路由器中，再进行故障处理。本实验较全面地介绍了 RIP 的排障流程，适合大部分 RIP 网络。

实验拓扑

公司网络拓扑如图 7-18 所示。

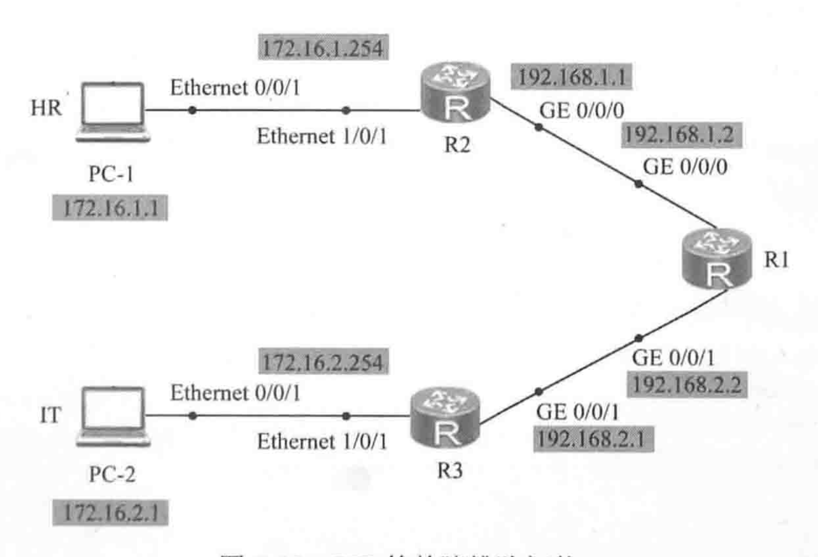


图 7-18　RIP 的故障排除拓扑

实验编址

实验编址见表 7-9。

表 7-9 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR1220）	GE 0/0/0	192.168.1.2	255.255.255.0	N/A
	GE 0/0/1	192.168.2.2	255.255.255.0	N/A
R2（AR1220）	GE 0/0/0	192.168.1.1	255.255.255.0	N/A
	Ethernet 1/0/1	172.16.1.254	255.255.255.0	N/A
R3（AR1220）	GE 0/0/1	192.168.2.1	255.255.255.0	N/A
	Ethernet 1/0/1	172.16.2.254	255.255.255.0	N/A
PC-1	Ethernet 0/0/1	172.16.1.1	255.255.255.0	172.16.1.254
PC-2	Ethernet 0/0/1	172.16.2.1	255.255.255.0	172.16.2.254

实验步骤

1. 导入设备预配置

本实验中设置了如下故障点：

- R3 缺少 **network** 192.168.2.0 命令；
- 在 R3 的 GE 0/0/1 接口下配置**undo rip input**命令；
- 关闭 R1 的 GE 0/0/1 接口；
- 在 R1 的 GE 0/0/0 接口下配置 **rip metricin 15** 命令；
- 在 R2 的 GE 0/0/0 接口下配置 RIP 认证，方式为明文认证，密码 huawei。

下面即为 3 台路由器 R1、R2、R3 的初始配置，直接使用即可。

```
system-view
sysname R1
interface GigabitEthernet0/0/0
ip address 192.168.1.2 255.255.255.0
rip metricin 15
interface GigabitEthernet0/0/1
ip address 192.168.2.2 255.255.255.0
shutdown
rip 1
version 2
network 192.168.1.0
network 192.168.2.0

system-view
sysname R2
interface Ethernet1/0/1
ip address 172.16.1.254 255.255.255.0
interface GigabitEthernet0/0/0
ip address 192.168.1.1 255.255.255.0
rip authentication-mode simple huawei
rip 1
version 2
network 172.16.0.0
network 192.168.1.0
```

```
system-view
sysname R3
interface Ethernet1/0/1
ip address 172.16.2.254 255.255.255.0
interface GigabitEthernet0/0/1
ip address 192.168.2.1 255.255.255.0
undo rip input
rip 1
version 2
network 172.16.0.0
```

在接下来的步骤中网络管理员需要根据逻辑思路，并借助必要的查看命令来进行排障。

2. 排除 R1 与 R2 间的故障

网络管理员发现公司网络此时出现故障，IT 部门与 HR 部门的终端无法正常互访。

测试 IT 部门 PC-1 与 HR 部门 PC-2 间的连通性。

```
PC>ping 172.16.2.1
Ping 172.16.2.1: 32 data bytes, Press Ctrl_C to break
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
--- 172.16.2.1 ping statistics ---
  5 packet(s) transmitted
  0 packet(s) received
  100.00% packet loss
```

可以观察到无法正常通信，即网络中存在故障，但此时网络管理员并不能确定故障位置。首先在 PC-1 上测试与网关设备 R2 间的连通性。

```
PC>ping 172.16.1.254
Ping 172.16.1.254: 32 data bytes, Press Ctrl_C to break
From 172.16.1.254: bytes=32 seq=1 ttl=255 time=15 ms
From 172.16.1.254: bytes=32 seq=2 ttl=255 time<1 ms
From 172.16.1.254: bytes=32 seq=3 ttl=255 time<1 ms
From 172.16.1.254: bytes=32 seq=4 ttl=255 time=16 ms
From 172.16.1.254: bytes=32 seq=5 ttl=255 time=16 ms
--- 172.16.1.254 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 0/9/16 ms
```

通信正常，表明 PC-1 跟网关设备 R2 间的链路没有问题。

路由器是根据路由表来进行数据转发的，所以在 R2 上使用 **display ip routing-table** 命令查看是否有 PC-2 所在网段的路由条目。

```
[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
Destinations : 6        Routes : 6
Destination/Mask    Proto    Pre    Cost    Flags NextHop     Interface
127.0.0.0/8         Direct   0      0       D     127.0.0.1   InLoopBack0
```

```
127.0.0.1/32        Direct  0   0   D   127.0.0.1       InLoopBack0
172.16.1.0/24       Direct  0   0   D   172.16.1.254    Ethernet1/0/1
172.16.1.254/32     Direct  0   0   D   127.0.0.1       Ethernet1/0/1
192.168.1.0/24      Direct  0   0   D   192.168.1.1     GigabitEthernet0/0/0
192.168.1.1/32      Direct  0   0   D   127.0.0.1       GigabitEthernet0/0/0
```

可以观察到，R2 上没有任何通过 RIP 协议接收的路由信息，说明 R1 与 R2 间的 RIP 路由信息通告不正常，即接下来需要在 R1 与 R2 之间排障。

第 1 步，检查 R1 与 R2 所在直连链路上的物理接口状态是否正常。

```
[R2]display ip interface brief GigabitEthernet 0/0/0
*down: administratively down
!down: FIB overload down
^down: standby
(l): loopback
(s): spoofing
(d): Dampening Suppressed
Interface                       IP Address/Mask     Physical    Protocol
GigabitEthernet0/0/0            192.168.1.1/24      up          up

[R1]display ip interface brief GigabitEthernet 0/0/0
*down: administratively down
!down: FIB overload down
^down: standby
(l): loopback
(s): spoofing
(d): Dampening Suppressed
Interface                       IP Address/Mask     Physical    Protocol
GigabitEthernet0/0/0            192.168.1.2/24      up          up
```

可以观察到，物理接口工作正常。“Physical”为 UP，即接口的物理状态处于正常启动的状态；“Protocol”为 UP，即接口的链路协议状态处于正常启动的状态。

在 R2 上测试与 R1 间直连链路的连通性。

```
[R2]ping -c 1 192.168.1.2
  PING 192.168.1.2: 56  data bytes, press CTRL_C to break
    Reply from 192.168.1.2: bytes=56 Sequence=1 ttl=255 time=70 ms
  --- 192.168.1.2 ping statistics ---
    1 packet(s) transmitted
    1 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 70/70/70 ms
```

可以观察到，连通性没有问题，继续下一步排查。

第 2 步，检查直连链路上的接口所在网段是否在 RIP 中通告。

```
[R2]display rip 1
Public VPN-instance
RIP process : 1
......
Verify-source : Enabled
Networks :
192.168.1.0         172.16.0.0
Configured peers                : None
......

[R1]display rip 1
Public VPN-instance
```

```
RIP process : 1
……
Verify-source : Enabled
Networks :
192.168.2.0          192.168.1.0
Configured peers                : None
……
```

可以观察到，接口的网段都已经在 RIP 中通告，继续下一步排查。

第 3 步，检查 R1、R2 上的 RIP 协议发送版本号和本地接口接收的版本号是否匹配。

在 R1、R2 上查看相应运行在 RIP 协议下的接口的详细信息。

```
[R1]display rip 1 interface GigabitEthernet 0/0/0 verbose
GigabitEthernet0/0/0(192.168.1.2)
State             : UP            MTU      : 500
Metricin          : 15
Metricout         : 1
Input             : Enabled       Output : Enabled
Protocol          : RIPv2 Multicast
Send version      : RIPv2 Multicast Packets
Receive version : RIPv2 Multicast and Broadcast Packets
Poison-reverse                    : Disabled
Split-Horizon                     : Enabled
Authentication type               : Simple
Replay Protection                 : Disabled

[R2]display rip 1 interface GigabitEthernet 0/0/0 verbose
GigabitEthernet0/0/0(192.168.1.1)
State             : UP            MTU      : 500
Metricin          : 0
Metricout         : 1
Input             : Enabled       Output : Enabled
Protocol          : RIPv2 Multicast
Send version      : RIPv2 Multicast Packets
Receive version : RIPv2 Multicast and Broadcast Packets
Poison-reverse                    : Disabled
Split-Horizon                     : Enabled
Authentication type               : None
Replay Protection                 : Disabled
```

可以观察到，双方的发送版本号和本地接口接收的版本号匹配，继续下一步排查。

第 4 步，由于目前在 R2 上没有接收到 R1 发送过来的路由信息，所以在 R2 上的入接口检查是否配置了**undo rip input**、**silent-interface**命令。

查看 R2 入接口 GE 0/0/0 的配置信息。

```
[R2]display current-configuration interface GigabitEthernet 0/0/0
#
interface GigabitEthernet0/0/0
ip address 192.168.1.1 255.255.255.0
rip authentication-mode simple cipher %$%$B""x"|=aa$$Ys*OM$OA&*pSf%$%$
#
Return
```

可以观察到，目前 R2 的入接口 GE 0/0/0 上并没有配置 **undo rip input**、**silent-interface** 命令。

第 5 步，检查是否在 RIP 进程中配置了 filter-policy 策略，来过滤掉收到的 RIP 路由

或不允许发送 RIP 路由。

```
[R2]rip
[R2-rip-1]display this
#
rip 1
version 2
network 172.16.0.0
network 192.168.1.0

[R1]rip
[R1-rip-1]display this
#
rip 1
version 2
network 192.168.1.0
network 192.168.2.0
```

可以观察到，并没有配置策略，继续下一步排查。

第 6 步，检查接口上是否已经开启水平分割，水平分割为默认开启。查看 R1 和 R2 相应接口上的 RIP 详细信息。

```
[R2]display rip 1 interface GigabitEthernet 0/0/0 verbose
 GigabitEthernet0/0/0(192.168.1.1)
......
  Poison-reverse              : Disabled
Split-Horizon                 : Enabled
Authentication type           : None
  Replay Protection           : Disabled

[R1]display rip 1 interface GigabitEthernet 0/0/0 verbose
GigabitEthernet0/0/0(192.168.1.2)
......
  Poison-reverse              : Disabled
Split-Horizon                 : Enabled
  Authentication type         : None
  Replay Protection           : Disabled
```

可以观察到，接口下没有关闭水平分割，继续下一步排查。

第 7 步，检查链路两端的接口认证方式是否匹配。使用 **display rip 1 statistics interface** 命令查看两端 RIP 接口上的统计信息。

```
[R1]display rip 1 statistics   interface GigabitEthernet 0/0/0
GigabitEthernet0/0/0(192.168.1.2)
Statistical information        Last min      Last 5 min      Total
------------------------------------------------------------------------
Periodic updates sent          1             6               361
......
Bad routes received            0             0               0
Packet authentication failed   0             3               168
Packet send failed             0             0               0

[R2]display rip 1 statistics interface GigabitEthernet 0/0/0
GigabitEthernet0/0/0(192.168.1.1)
Statistical information        Last min      Last 5 min      Total
------------------------------------------------------------------------
Periodic updates sent          0             4               361
```

```
......
Bad routes received            0          0          0
Packet authentication failed   0          4          328
Packet send failed             0          0          0
```

观察到有认证失败的 RIP 报文，说明两端的 RIP 认证方式有问题。在双方路由器上执行 **display current-configuration** 命令查看配置信息。

```
[R2]display current-configuration
#
interface GigabitEthernet0/0/0
 ip address 192.168.1.1 255.255.255.0
 rip authentication-mode simple cipher %$%$B""x"|=aa$$Ys*OM$OA&*pSf%$%$

[R1]display current-configuration
#
interface GigabitEthernet0/0/0
 ip address 192.168.1.2 255.255.255.0
```

发现 R2 的 GE 0/0/0 接口下配置了 RIP 认证，而 R1 的 GE 0/0/0 接口下没有配置认证。

进入 R2 的 GE 0/0/0 接口删除该认证命令。

```
[R2]interface GigabitEthernet0/0/0
[R2-GigabitEthernet0/0/0]undo rip authentication-mode
```

配置完成后，使用 **display ip routing-table protocol rip** 命令检查双方现在是否能够正常收发 RIP 路由。

```
[R2]display ip routing-table protocol rip
[R2]

[R1]display ip routing-table protocol rip
[R1]
```

发现仍然没有相关的 RIP 路由条目，继续下一步排查。

第 8 步，查看路由表中是否存在从其他协议获得的相同路由，查看 R2 的路由表信息。

```
[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
Destinations : 6        Routes : 6
Destination/Mask   Proto    Pre   Cost   Flags NextHop        Interface
127.0.0.0/8        Direct   0     0      D    127.0.0.1       InLoopBack0
127.0.0.1/32       Direct   0     0      D    127.0.0.1       InLoopBack0
172.16.1.0/24      Direct   0     0      D    172.16.1.254    Ethernet1/0/1
172.16.1.254/32    Direct   0     0      D    127.0.0.1       Ethernet1/0/1
192.168.1.0/24     Direct   0     0      D    192.168.1.1     GigabitEthernet0/0/0
192.168.1.1/32     Direct   0     0      D    127.0.0.1       GigabitEthernet0/0/0
```

可以观察到，并没从有其他路由协议获得相同的路由，继续下一步排查。

第 9 步，检查收到的路由度量值是否大于 16。使用 **display rip 1 route** 命令查看。

```
[R1]display rip 1 route
Route Flags : R - RIP
A - Aging, G - Garbage-collect
------------------------------------------------------------------------------
Peer 192.168.1.1 on GigabitEthernet0/0/0
Destination/Mask   Nexthop       Cost   Tag   Flags   Sec
172.16.1.0/24      192.168.1.1   16     0     RG      16
```

发现从 R1 接收到的 172.16.1.0 网段的路由条目度量值是 16，该路由不可达，所以

不会将该路由添加到路由表中。注意，如果操作到该步骤时距离导入预配置的时间间隔过久，将会无法查看到该结果。

在 R1 上使用 **display current-configuration | include rip** 命令，查看包含字符串"rip"的所有配置信息。

```
[R1]display current-configuration | include rip
rip metricin 15
rip 1
```

发现 R1 上配置了 **rip metricin 15** 命令，将 GE 0/0/0 接口接收到的路由都加上了 15 的度量值，再放入路由表中，导致 172.16.1.0 网段的路由条目的度量值是 16。

进入 GE 0/0/0 接口，删除该命令。

```
[R1]interface GigabitEthernet0/0/0
[R1-GigabitEthernet0/0/0]undo rip metricin
```

配置完成后，在 R1 上使用 **display ip routing-table** 命令，检查路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
Destinations : 8        Routes : 8
Destination/Mask    Proto   Pre   Cost   Flags   NextHop       Interface
127.0.0.0/8         Direct  0     0      D       127.0.0.1     InLoopBack0
127.0.0.1/32        Direct  0     0      D       127.0.0.1     InLoopBack0
127.255.255.255/32  Direct  0     0      D       127.0.0.1     InLoopBack0
172.16.1.0/24       RIP     100   1      D       192.168.1.1   GigabitEthernet0/0/0
192.168.1.0/24      Direct  0     0      D       192.168.1.2   GigabitEthernet0/0/0
192.168.1.2/32      Direct  0     0      D       127.0.0.1     GigabitEthernet0/0/0
192.168.1.255/32    Direct  0     0      D       127.0.0.1     GigabitEthernet0/0/0
255.255.255.255/32  Direct  0     0      D       127.0.0.1     InLoopBack0
```

可以发现 R1 已经收到 172.16.1.0 网段的路由信息，接收到了 R2 通告的路由条目，即 R1 与 R2 之间的故障已排除完毕。

但是此时 R1 仍然没有 R3 上 172.16.2.0 网段的路由信息，说明 R1 与 R3 间的 RIP 路由信息通告不正常。接下来需要在 R1 与 R3 之间排障。

3. 排除 R1 与 R3 间的故障

第 1 步，检查 R1 与 R3 所在直连链路上的物理接口状态是否正常。

```
[R1]display ip interface brief
*down: administratively down
^down: standby
(l): loopback
(s): spoofing
The number of interface that is UP in Physical is 2
The number of interface that is DOWN in Physical is 2
The number of interface that is UP in Protocol is 2
The number of interface that is DOWN in Protocol is 2
Interface                 IP Address/Mask     Physical   Protocol
GigabitEthernet0/0/0      192.168.1.2/24      up         up
GigabitEthernet0/0/1      192.168.2.2/24      *down      down
NULL0                     unassigned          up         up(s)
```

可以观察到，物理接口工作不正常。接口的物理状态和链路协议状态都不正常，并且"Physical"状态为"*down"，表示网络管理员在该接口执行了 **shutdown** 命令。

进入 R1 的 GE 0/0/1 接口，通过 **display this** 命令查看接口配置。

```
[R1]interface GigabitEthernet 0/0/1
[R1-GigabitEthernet0/0/1]display this
[V200R003C00]
#
interface GigabitEthernet0/0/1
 shutdown
 ip address 192.168.2.2 255.255.255.0
```

果然发现接口下配置了 **shutdown** 命令。在该接口配置 **undo shutdown** 命令。

```
[R1-GigabitEthernet0/0/1]undo shutdown
```

配置完成后，测试连通性。

```
[R1]ping 192.168.2.1
PING 192.168.2.1: 56  data bytes, press CTRL_C to break
Reply from 192.168.2.1: bytes=56 Sequence=1 ttl=255 time=210 ms
Reply from 192.168.2.1: bytes=56 Sequence=2 ttl=255 time=60 ms
Reply from 192.168.2.1: bytes=56 Sequence=3 ttl=255 time=50 ms
Reply from 192.168.2.1: bytes=56 Sequence=4 ttl=255 time=20 ms
Reply from 192.168.2.1: bytes=56 Sequence=5 ttl=255 time=20 ms
--- 192.168.2.1 ping statistics ---
5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 20/72/210 ms
```

现在恢复正常连通。检查 R1 的路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8        Routes : 8
Destination/Mask    Proto   Pre   Cost   Flags  NextHop       Interface
 127.0.0.0/8        Direct  0     0      D      127.0.0.1     InLoopBack0
127.0.0.1/32        Direct  0     0      D      127.0.0.1     InLoopBack0
127.255.255.255/32  Direct  0     0      D      127.0.0.1     InLoopBack0
172.16.1.0/24       RIP     100   1      D      192.168.1.1   GigabitEthernet0/0/0
192.168.1.0/24      Direct  0     0      D      192.168.1.2   GigabitEthernet0/0/0
192.168.1.2/32      Direct  0     0      D      127.0.0.1     GigabitEthernet0/0/0
192.168.1.255/32    Direct  0     0      D      127.0.0.1     GigabitEthernet0/0/0
255.255.255.255/32  Direct  0     0      D      127.0.0.1     InLoopBack0
```

发现没有收到相关的 RIP 路由条目，继续下一步排查。

第 2 步，检查直连链路上的接口所在网段是否在 RIP 中通告。

```
[R1]display rip 1
Public VPN-instance
RIP process : 1
......
Verify-source : Enabled
Networks : 192.168.2.0          192.168.1.0
Configured peers                : None
......

[R3]display rip 1
Public VPN-instance
RIP process : 1
```

```
255.255.255.255/32   Direct   0     0     D   127.0.0.1      InLoopBack0

[R3]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 12        Routes : 12
Destination/Mask     Proto    Pre   Cost  Flags NextHop        Interface
127.0.0.0/8          Direct   0     0     D   127.0.0.1      InLoopBack0
127.0.0.1/32         Direct   0     0     D   127.0.0.1      InLoopBack0
127.255.255.255/32   Direct   0     0     D   127.0.0.1      InLoopBack0
172.16.1.0/24        RIP      100   2     D   192.168.2.2    GigabitEthernet0/0/1
172.16.2.0/24        Direct   0     0     D   172.16.2.254   Ethernet1/0/1
172.16.2.254/32      Direct   0     0     D   127.0.0.1      Ethernet1/0/1
172.16.2.255/32      Direct   0     0     D   127.0.0.1      Ethernet1/0/1
192.168.1.0/24       RIP      100   1     D   192.168.2.2    GigabitEthernet0/0/1
192.168.2.0/24       Direct   0     0     D   192.168.2.1    GigabitEthernet0/0/1
192.168.2.1/32       Direct   0     0     D   127.0.0.1      GigabitEthernet0/0/1
192.168.2.255/32     Direct   0     0     D   127.0.0.1      GigabitEthernet0/0/1
255.255.255.255/32   Direct   0     0     D   127.0.0.1      InLoopBack0
```

可以观察到，双方能正常接收到对方 RIP 路由。

最后在 PC-1 上测试与 PC-2 间的连通性。

```
PC>ping 172.16.2.1
Ping 172.16.2.1: 32 data bytes, Press Ctrl_C to break
From 172.16.2.1: bytes=32 seq=1 ttl=125 time=93 ms
From 172.16.2.1: bytes=32 seq=2 ttl=125 time=31 ms
From 172.16.2.1: bytes=32 seq=3 ttl=125 time=16 ms
From 172.16.2.1: bytes=32 seq=4 ttl=125 time=16 ms
From 172.16.2.1: bytes=32 seq=5 ttl=125 time=15 ms
--- 172.16.2.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 15/34/93 ms
```

通信正常，至此故障排除完成。

思考

如果采用 debug 或者抓包的方式排错，与采用查看命令进行排错相比有什么优劣？

7.10 RIP 的路由引入

原理概述

设计者在进行网络规划或设计时，一般都设计成仅运行一种路由协议，以降低网络复杂性，易于维护。但是，如果在网络升级、扩展或合并时，就可能造成在网络中同时运行几种不同的路由协议，这时就需要部署路由协议间的引入，使路由信息能够在不同协议间传递。

RIP 支持不同路由协议的引入，包括直连路由、静态路由或其他动态路由协议。由

于 RIP 的度量值是跳数且最大值不能超过 15，所以在将其他路由协议引入至 RIP 时需要注意设置度量值，避免引入的路由度量值超过 15。默认情况下，引入另一种协议或引入同种协议的不同进程时往往是把该协议或该进程的所有路由一起引入，可以在引入的同时通过设置策略来控制和过滤特定的路由信息。

实验目的

- 掌握 RIP 路由引入的应用场景
- 掌握在 RIP 中引入直连路由的配置
- 掌握在 RIP 中引入静态路由的配置
- 理解 RIP 抑制接口的使用场景

实验内容

A 和 B 两家公司，R4 是公司 A 的网关路由器，左侧连接的公司 A 内网；R1 是公司 B 的网关路由器，右侧是公司 B 的内网。内网中的 R2 连接财务部门，R3 连接研发部门，3 台路由器运行 RIP 协议。财务部门和研发部门不希望接收到大量 RIP 的更新报文，通过把它们的网段当作外部网络引入到 RIP 中来实现。而在优化完公司 B 的 RIP 网络之后，要求公司 B 与公司 A 能够互相通信，现需要使用静态路由和路由引入技术使两家公司的网络能够互访。

实验拓扑

RIP 的路由引入拓扑如图 7-19 所示。

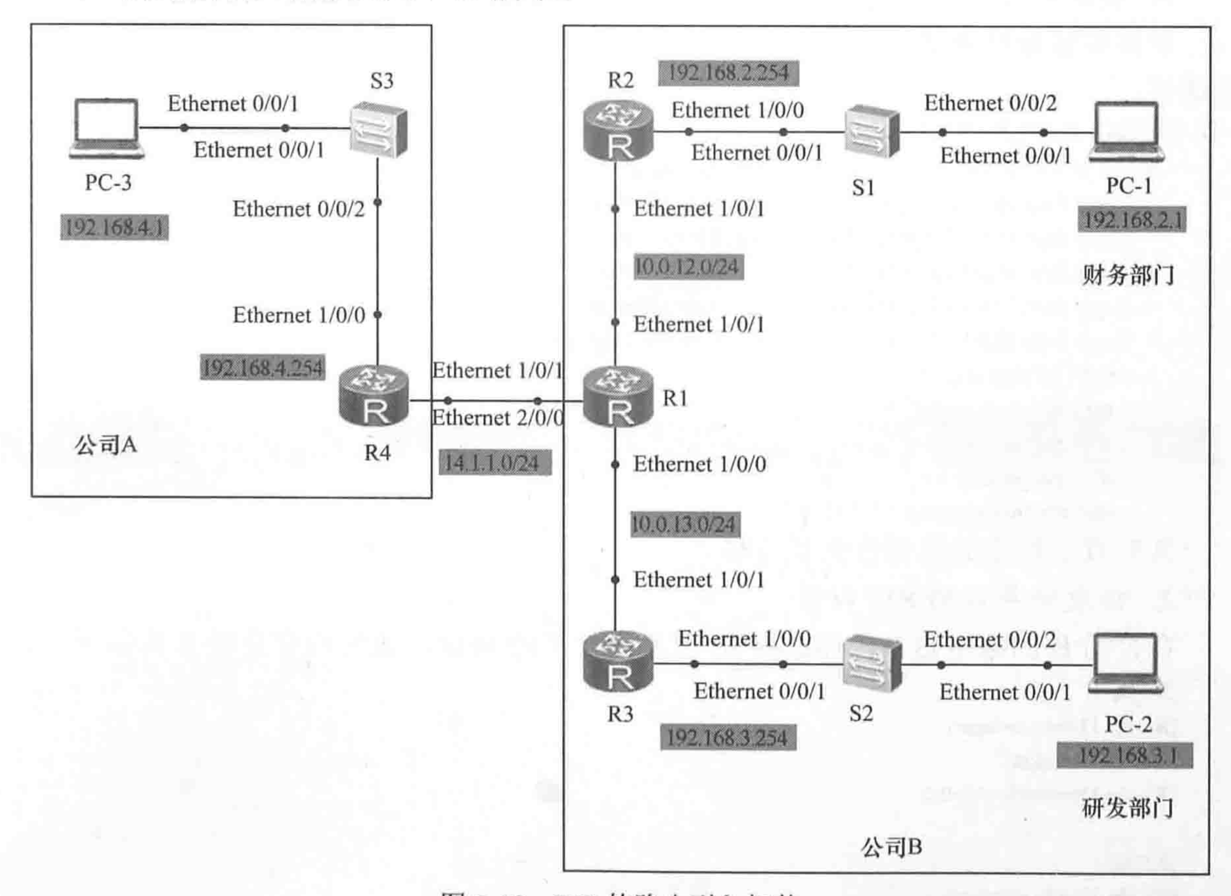


图 7-19　RIP 的路由引入拓扑

实验编址

实验编址见表 7-10。

表 7-10 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR1220）	Ethernet 1/0/0	10.0.13.1	255.255.255.0	N/A
	Ethernet 1/0/1	10.0.12.1	255.255.255.0	N/A
	Ethernet 2/0/0	14.1.1.1	255.255.255.0	N/A
R2（AR1220）	Ethernet 1/0/0	192.168.2.254	255.255.255.0	N/A
	Ethernet 1/0/1	10.0.12.2	255.255.255.0	N/A
R3（AR1220）	Ethernet 1/0/0	192.168.3.254	255.255.255.0	N/A
	Ethernet 1/0/1	10.0.13.3	255.255.255.0	N/A
R4（AR1220）	Ethernet 1/0/1	14.1.1.4	255.255.255.0	N/A
	Ethernet 1/0/0	192.168.4.254	255.255.255.0	N/A
PC-1	Ethernet 0/0/1	192.168.2.1	255.255.255.0	192.168.2.254
PC-2	Ethernet 0/0/1	192.168.3.1	255.255.255.0	192.168.3.254
PC-3	Ethernet 0/0/1	192.168.4.1	255.255.255.0	192.168.4.254

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 地址配置，并使用 **ping** 命令检测各直连链路的连通性。

```
[R1]ping 10.0.12.2
  PING 10.0.12.2: 56   data bytes, press CTRL_C to break
    Reply from 10.0.12.2: bytes=56 Sequence=1 ttl=255 time=10 ms
    Reply from 10.0.12.2: bytes=56 Sequence=2 ttl=255 time=1 ms
    Reply from 10.0.12.2: bytes=56 Sequence=3 ttl=255 time=1 ms
    Reply from 10.0.12.2: bytes=56 Sequence=4 ttl=255 time=1 ms
    Reply from 10.0.12.2: bytes=56 Sequence=5 ttl=255 time=1 ms
  --- 10.0.12.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 1/2/10 ms
```

其余直连网段的连通性测试省略。

2. 搭建公司 B 的 RIP 网络

在公司 B 的路由器 R1、R2 和 R3 上配置 RIPv2 协议，通告所有公司 B 内部网段。

```
[R1]rip 1
[R1-rip-1]undo summary
[R1-rip-1]version 2
[R1-rip-1]network 10.0.0.0

[R2]rip 1
[R2-rip-1]undo summary
```

```
[R2-rip-1]version 2
[R2-rip-1]network 10.0.0.0
[R2-rip-1]network 192.168.2.0

[R3-rip-1]rip 1
[R3-rip-1]undo summary
[R3-rip-1]version 2
[R3-rip-1]network 10.0.0.0
[R3-rip-1]network 192.168.3.0
```

配置完成后，查看 R1 路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 15        Routes : 15
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      10.0.12.0/24  Direct  0    0           D   10.0.12.1       Ethernet1/0/1
      10.0.12.1/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/1
    10.0.12.255/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/1
      10.0.13.0/24  Direct  0    0           D   10.0.13.1       Ethernet1/0/0
      10.0.13.1/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/0
    10.0.13.255/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/0
       14.1.1.0/24  Direct  0    0           D   14.1.1.1        Ethernet2/0/0
       14.1.1.1/32  Direct  0    0           D   127.0.0.1       Ethernet2/0/0
     14.1.1.255/32  Direct  0    0           D   127.0.0.1       Ethernet2/0/0
        127.0.0.0/8 Direct  0    0           D   127.0.0.1       InLoopBack0
       127.0.0.1/32 Direct  0    0           D   127.0.0.1       InLoopBack0
127.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0
     192.168.2.0/24 RIP     100  1           D   10.0.12.2       Ethernet1/0/1
     192.168.3.0/24 RIP     100  1           D   10.0.13.3       Ethernet1/0/0
255.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0
```

可以观察到，此时公司 B 的网关路由器 R1 已经成功接收到了内网中财务部 192.168.2.0/24 和研发部门 192.168.3.0/24 网段的路由条目。

3. 优化公司 B 的 RIP 网络

公司 B 网络搭建完成后，网络管理员对网络进行维护。在 R2 的 E 1/0/0 接口抓取数据包，如图 7-20 所示。

```
No.  Time          Source          Destination   Protocol Info
  9 15.616000      192.168.2.254   224.0.0.9     RIPv2    Response
 23 43.571000      192.168.2.254   224.0.0.9     RIPv2    Response
 37 73.539000      192.168.2.254   224.0.0.9     RIPv2    Response

⊞ Frame 37: 126 bytes on wire (1008 bits), 126 bytes captured (1008 bits)
⊞ Ethernet II, Src: HuaweiTe_04:bf:7d (00:e0:fc:04:bf:7d), Dst: IPv4mcast_00:00:09 (01:00:5e:00:00:09)
⊞ Internet Protocol, Src: 192.168.2.254 (192.168.2.254), Dst: 224.0.0.9 (224.0.0.9)
⊞ User Datagram Protocol, Src Port: router (520), Dst Port: router (520)
⊟ Routing Information Protocol
    Command: Response (2)
    Version: RIPv2 (2)
    Routing Domain: 0
  ⊞ IP Address: 10.0.12.0, Metric: 1
  ⊞ IP Address: 10.0.13.0, Metric: 2
  ⊞ IP Address: 192.168.2.0, Metric: 1
  ⊞ IP Address: 192.168.3.0, Metric: 3
```

图 7-20 抓包观察

可以观察到，此时 R2 上连接财务部门终端一侧的接口上会通告 RIP 路由信息，而这些 RIP 报文对终端 PC 而言是毫无用处的。原因是使用 **network** 命令通告财务部门所在网段后，R2 的该 E 1/0/0 接口就会收发 RIP 协议报文，不管对端设备是否利用。

为了使财务部门的终端不接收这些无用 RIP 更新报文，可以在 R2 的 RIP 进程中不使用 **network** 命令通告该网段，而采用引入直连路由的方式来代替，将财务部门的网段作为外部路由发布到公司 RIP 网络中。

在 R2 上使用 **import-route** 命令配置路由引入，指定引入的源路由协议为直连路由。

提示 在一台设备上配置路由引入时，需要保证被引入的路由条目已经存在于当前设备的路由表中。

```
[R2]rip 1
[R2-rip-1]undo network 192.168.2.0
[R2-rip-1]import-route direct
```

配置完成后，查看 R1 路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 15        Routes : 15
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      10.0.12.0/24  Direct  0    0           D   10.0.12.1       Ethernet1/0/1
      10.0.12.1/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/1
    10.0.12.255/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/1
      10.0.13.0/24  Direct  0    0           D   10.0.13.1       Ethernet1/0/0
      10.0.13.1/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/0
    10.0.13.255/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/0
       14.1.1.0/24  Direct  0    0           D   14.1.1.1        Ethernet2/0/0
       14.1.1.1/32  Direct  0    0           D   127.0.0.1       Ethernet2/0/0
     14.1.1.255/32  Direct  0    0           D   127.0.0.1       Ethernet2/0/0
       127.0.0.0/8  Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
127.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0
    192.168.2.0/24  RIP     100  1           D   10.0.12.2       Ethernet1/0/1
    192.168.3.0/24  RIP     100  1           D   10.0.13.3       Ethernet1/0/0
255.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0
```

可以观察到，R1 上接收到了 R2 引入的 192.168.2.0/24 网段的路由信息。

再次在 R2 的 E 1/0/0 接口下抓包，如图 7-21 所示。

可以观察到，现在该接口上没有发送任何 RIP 更新报文，即此时已经完成优化，财务部门的终端不再收到与其无关的 RIP 更新报文。

可以在 R2 的 E 1/0/1 接口上抓取数据包观察区别，如图 7-22 所示。

可以观察到，在该 E 1/0/1 的接口上仍然正常发送 RIP 更新报文，将引入后的 192.168.2.0/24 网段通告出去。

研发部门也会有相同的问题——收到对用户无用的 RIP 报文，同样采用引入直连路由的方式来解决，此处省略。

```
No.  Time      Source             Destination        Protocol Info
382 812.1880 HuaweiTe_79:41: Spanning-tree-( STP   MST. Root = 32768/0/4c:1f:cc:7
383 814.4040 HuaweiTe_79:41: Spanning-tree-( STP   MST. Root = 32768/0/4c:1f:cc:7
384 816.6340 HuaweiTe_79:41: Spanning-tree-( STP   MST. Root = 32768/0/4c:1f:cc:7
385 818.7720 HuaweiTe_79:41: Spanning-tree-( STP   MST. Root = 32768/0/4c:1f:cc:7
386 820.9090 HuaweiTe_79:41: Spanning-tree-( STP   MST. Root = 32768/0/4c:1f:cc:7
387 823.0930 HuaweiTe_79:41: Spanning-tree-( STP   MST. Root = 32768/0/4c:1f:cc:7
388 825.2300 HuaweiTe_79:41: Spanning-tree-( STP   MST. Root = 32768/0/4c:1f:cc:7
389 827.3830 HuaweiTe_79:41: Spanning-tree-( STP   MST. Root = 32768/0/4c:1f:cc:7
390 829.5510 HuaweiTe_79:41: Spanning-tree-( STP   MST. Root = 32768/0/4c:1f:cc:7
391 831.7820 HuaweiTe_79:41: Spanning-tree-( STP   MST. Root = 32768/0/4c:1f:cc:7
392 833.9190 HuaweiTe_79:41: Spanning-tree-( STP   MST. Root = 32768/0/4c:1f:cc:7
393 836.0410 HuaweiTe_79:41: Spanning-tree-( STP   MST. Root = 32768/0/4c:1f:cc:7
394 838.1630 HuaweiTe_79:41: Spanning-tree-( STP   MST. Root = 32768/0/4c:1f:cc:7

⊞ Frame 100: 119 bytes on wire (952 bits), 119 bytes captured (952 bits)
⊞ IEEE 802.3 Ethernet
⊞ Logical-Link Control
⊟ Spanning Tree Protocol
    Protocol Identifier: Spanning Tree Protocol (0x0000)
    Protocol Version Identifier: Multiple Spanning Tree (3)
    BPDU Type: Rapid/Multiple Spanning Tree (0x02)
  ⊞ BPDU flags: 0x7c (Agreement, Forwarding, Learning, Port Role: Designated)
  ⊞ Root Identifier: 32768 / 0 / 4c:1f:cc:79:41:39
    Root Path Cost: 0
  ⊞ Bridge Identifier: 32768 / 0 / 4c:1f:cc:79:41:39
    Port identifier: 0x8001
    Message Age: 0
    Max Age: 20

0000  01 80 c2 00 00 00 4c 1f  cc 79 41 39 00 69 42 42   ......L. .yA9.iBB
0010  03 00 00 03 02 7c 80 00  4c 1f cc 79 41 39 00 00   .....|.. L..yA9..
0020  00 00 80 00 4c 1f cc 79  41 39 80 01 00 00 14 00   ....L..y A9......
0030  02 00 0f 00 00 00 40 00  34 63 31 66 63 63 37 39   ......@. 4c1fcc79
```

图 7-21　抓包观察

```
No.  Time        Source       Destination    Protocol Info
  4 57.174000  10.0.12.2    224.0.0.9      RIPv2   Response
  5 60.029000  10.0.12.1    224.0.0.9      RIPv2   Response
  6 90.043000  10.0.12.1    224.0.0.9      RIPv2   Response
  7 92.134000  10.0.12.2    224.0.0.9      RIPv2   Response

⊞ Frame 7: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)
⊞ Ethernet II, Src: HuaweiTe_04:bf:7e (00:e0:fc:04:bf:7e), Dst: IPv4mcast_00:00:09 (01:00:5e:00:00:09)
⊞ Internet Protocol, Src: 10.0.12.2 (10.0.12.2), Dst: 224.0.0.9 (224.0.0.9)
⊞ User Datagram Protocol, Src Port: router (520), Dst Port: router (520)
⊟ Routing Information Protocol
    Command: Response (2)
    Version: RIPv2 (2)
    Routing Domain: 0
  ⊟ IP Address: 192.168.2.0, Metric: 1
      Address Family: IP (2)
      Route Tag: 0
      IP Address: 192.168.2.0 (192.168.2.0)
      Netmask: 255.255.255.0 (255.255.255.0)
      Next Hop: 0.0.0.0 (0.0.0.0)
      Metric: 1
```

图 7-22　抓包观察

4. 连接公司 A 与公司 B 的网络

由于业务需要，需要公司 A 与 B 的网络能够互相访问。

在公司 B 的网关设备 R1 上配置目的为 192.168.4.0/24 网段的静态路由，并在 RIP 进程中引入该条静态路由，引入后公司 B 中 RIP 网络内的所有路由器会通过 RIP 协议自动学习到该路由。

```
[R1]ip route-static 192.168.4.0 255.255.255.0 14.1.1.4
[R1]rip 1
[R1-rip-1]import-route static
```

配置完成后，查看公司 B 的 R2、R3 路由表。

```
<R2>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 13        Routes : 13
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      10.0.12.0/24  Direct  0    0           D   10.0.12.2       Ethernet1/0/1
      10.0.12.2/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/1
    10.0.12.255/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/1
      10.0.13.0/24  RIP     100  1           D   10.0.12.1       Ethernet1/0/1
       127.0.0.0/8  Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
127.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0
    192.168.2.0/24  Direct  0    0           D   192.168.2.254   Ethernet1/0/0
  192.168.2.254/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/0
  192.168.2.255/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/0
    192.168.3.0/24  RIP     100  2           D   10.0.12.1       Ethernet1/0/1
    192.168.4.0/24  RIP     100  1           D   10.0.12.1       Ethernet1/0/1
255.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0

<R3>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 13        Routes : 13
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      10.0.12.0/24  RIP     100  1           D   10.0.13.1       Ethernet1/0/1
      10.0.13.0/24  Direct  0    0           D   10.0.13.3       Ethernet1/0/1
      10.0.13.3/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/1
    10.0.13.255/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/1
       127.0.0.0/8  Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
127.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0
    192.168.2.0/24  RIP     100  2           D   10.0.13.1       Ethernet1/0/1
    192.168.3.0/24  Direct  0    0           D   192.168.3.254   Ethernet1/0/0
  192.168.3.254/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/0
  192.168.3.255/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/0
    192.168.4.0/24  RIP     100  1           D   10.0.13.1       Ethernet1/0/1
255.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0
```

可以观察到，此时公司 B 的内部路由器 R2 和 R3 上能够正常获得公司 A 内部网段的路由信息。但是此时公司 A 的路由器上仍然还没有公司 B 的任何路由信息。

在 R4 上配置一条默认路由，下一跳为 R1。

```
[R4]ip route-static 0.0.0.0 0.0.0.0 14.1.1.1
```

配置完成后，查看 R4 路由表。

```
[R4]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 11        Routes : 11
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
         0.0.0.0/0  Static  60   0          RD   14.1.1.1        Ethernet1/0/1
        14.1.1.0/24 Direct  0    0           D   14.1.1.4        Ethernet1/0/1
        14.1.1.4/32 Direct  0    0           D   127.0.0.1       Ethernet1/0/1
```

```
      14.1.1.255/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/1
        127.0.0.0/8  Direct  0    0           D   127.0.0.1       InLoopBack0
       127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
 127.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0
     192.168.4.0/24  Direct  0    0           D   192.168.4.254   Ethernet1/0/0
   192.168.4.254/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/0
   192.168.4.255/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/0
255.255.255.255/32   Direct  0    0           D   127.0.0.1       InLoopBack0
```

可以观察到，静态路由配置成功。

在 PC-1 上测试与 PC-3 间的连通性。

```
PC>ping 192.168.4.1
Ping 192.168.4.1: 32 data bytes, Press Ctrl_C to break
From 192.168.4.1: bytes=32 seq=1 ttl=125 time=47 ms
From 192.168.4.1: bytes=32 seq=2 ttl=125 time=62 ms
From 192.168.4.1: bytes=32 seq=3 ttl=125 time=63 ms
From 192.168.4.1: bytes=32 seq=4 ttl=125 time=47 ms
From 192.168.4.1: bytes=32 seq=5 ttl=125 time=63 ms
--- 192.168.4.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 47/56/63 ms
```

可以观察到，PC-1 与 PC-3 通信正常。至此，公司 A 和 B 之间可以正常通信。

思考

使用 **network** 命令方式通告路由，与路由引入的方式通告路由有什么区别？

第8章 OSPF

8.1 OSPF单区域配置

8.2 OSPF多区域配置

8.3 配置OSPF的认证

8.4 OSPF被动接口配置

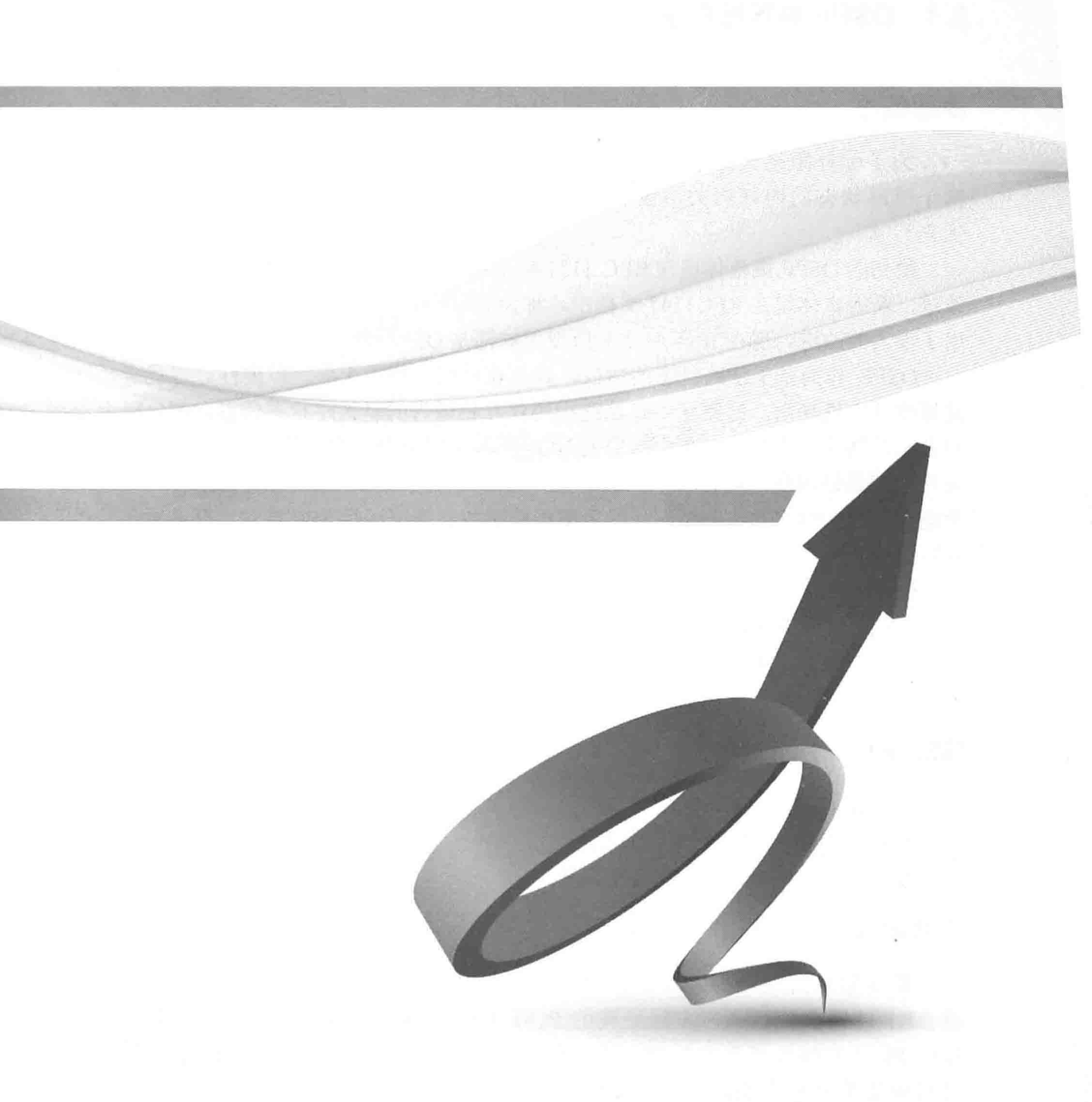
8.5 理解OSPF Router-ID

8.6 OSPF的DR与BDR

8.7 OSPF开销值、协议优先级及计时器的修改

8.8 连接RIP与OSPF网络

8.9 使用RIP、OSPF发布默认路由



8.1 OSPF 单区域配置

原理概述

为了弥补距离矢量路由协议的不足，IETF 组织于 20 世纪 80 年代末开发了一种基于链路状态的内部网关协议——OSPF（Open Shortest Path First，开放式最短路径优先）。

最初的 OSPF 规范体现在 RFC 1131 中，这个第 1 版（OSPFv1）很快被进行了重大改进，新版本体现在 RFC1247 文档中，称为 OSPFv2，版本 2 在稳定性和功能性方面做出了很大的改进。现在 IPv4 网络中所使用的都是 OSPFv2。

OSPF 作为基于链路状态的协议，具有收敛快、路由无环、扩展性好等优点，被快速接受并广泛使用。链路状态算法路由协议互相通告的是链路状态信息，每台路由器都将自己的链路状态信息（包含接口的 IP 地址和子网掩码、网络类型、该链路的开销等）发送给其他路由器，并在网络中泛洪，当每台路由器收集到网络内所有链路状态信息后，就能拥有整个网络的拓扑情况，然后根据整网拓扑情况运行 SPF 算法，得出所有网段的最短路径。

OSPF 支持区域的划分，区域是从逻辑上将路由器划分为不同的组，每个组用区域号（Area ID）来标识。一个网段（链路）只能属于一个区域，或者说每个运行 OSPF 的接口必须指明属于哪一个区域。区域 0 为骨干区域，骨干区域负责在非骨干区域之间发布区域间的路由信息。在一个 OSPF 区域中有且只有一个骨干区域。

实验目的

- 掌握 OSPF 单区域的配置方法
- 理解 OSPF 单区域的应用场景
- 掌握查看 OSPF 邻居状态的方法

实验内容

本实验模拟企业网络场景。该公司有三大办公区，每个办公区放置了一台路由器，R1 放在办公区 A，A 区经理的 PC-1 直接连接 R1；R2 放在办公区 B，B 区经理的 PC-2 直接连接到 R2；R3 放在办公区 C，C 区经理的 PC-3 直接连接到 R3；3 台路由器都互相直连，为了能使整个公司网络互相通信，需要在所有路由器上部署路由协议。考虑到公司未来的发展（部门的增加和分公司的成立），为了适应不断扩展的网络的需求，公司在所有路由器上部署 OSPF 协议，且现在所有路由器都属于骨干区域。

实验拓扑

OSPF 单区域配置的拓扑如图 8-1 所示。

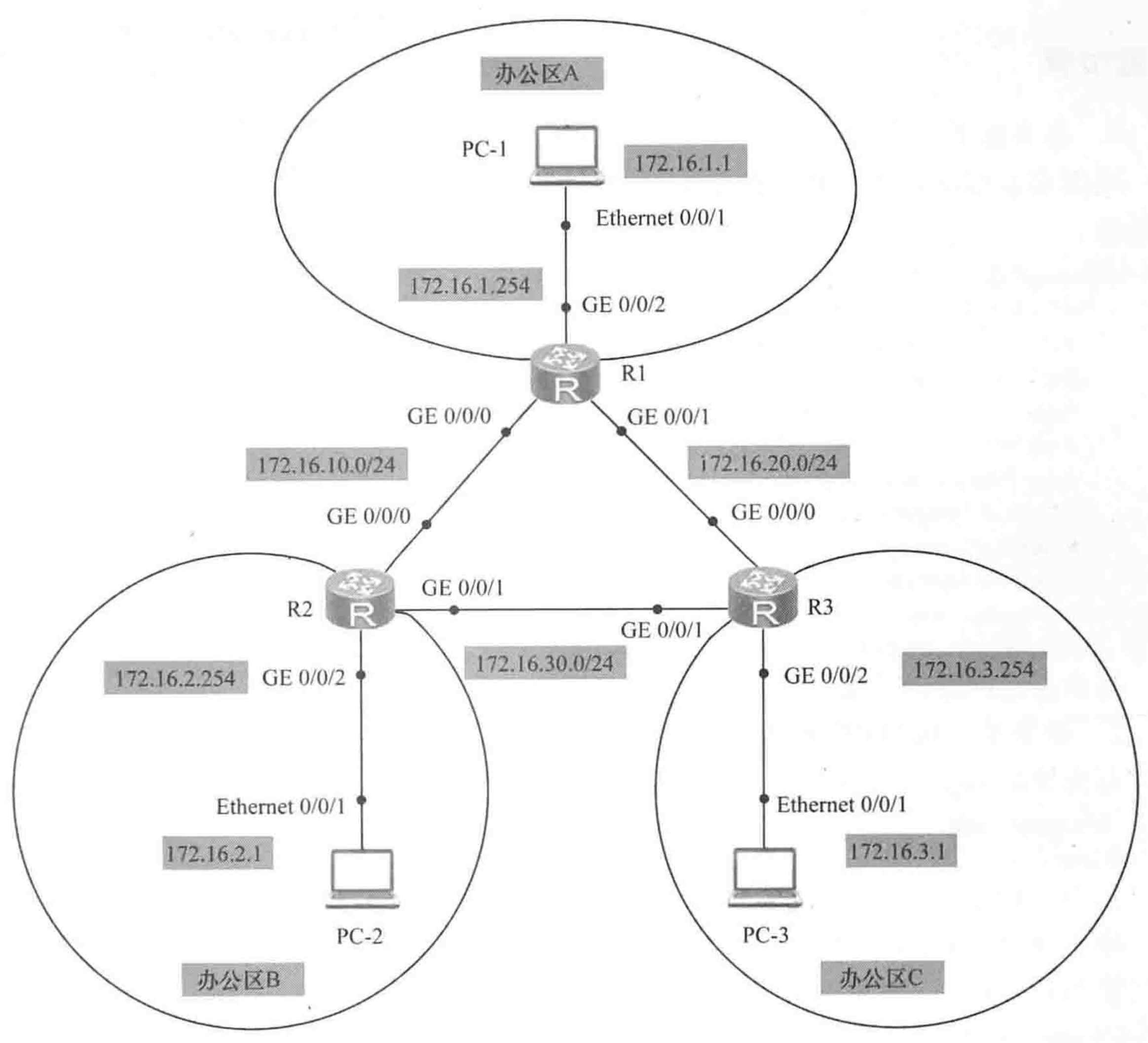


图 8-1　OSPF 单区域配置拓扑

实验编址

实验编址见表 8-1。

表 8-1　　实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR2220）	GE 0/0/0	172.16.10.1	255.255.255.0	N/A
	GE 0/0/1	172.16.20.1	255.255.255.0	N/A
	GE 0/0/2	172.16.1.254	255.255.255.0	N/A
R2（AR2220）	GE 0/0/0	172.16.10.2	255.255.255.0	N/A
	GE 0/0/1	172.16.30.2	255.255.255.0	N/A
	GE 0/0/2	172.16.2.254	255.255.255.0	N/A
R3（AR2220）	GE 0/0/0	172.16.20.3	255.255.255.0	N/A
	GE 0/0/1	172.16.30.3	255.255.255.0	N/A
	GE 0/0/2	172.16.3.254	255.255.255.0	N/A
PC-1	Ethernet 0/0/1	172.16.1.1	255.255.255.0	172.16.1.254
PC-2	Ethernet 0/0/1	172.16.2.1	255.255.255.0	172.16.2.254
PC-3	Ethernet 0/0/1	172.16.3.1	255.255.255.0	172.16.3.254

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 地址配置，并使用 **ping** 命令检测各直连链路的连通性。

```
<R1>ping 172.16.20.3
  PING 172.16.20.3: 56    data bytes, press CTRL_C to break
    Reply from 172.16.20.3: bytes=56 Sequence=1 ttl=255 time=20 ms
    Reply from 172.16.20.3: bytes=56 Sequence=2 ttl=255 time=20 ms
    Reply from 172.16.20.3: bytes=56 Sequence=3 ttl=255 time=10 ms
    Reply from 172.16.20.3: bytes=56 Sequence=4 ttl=255 time=10 ms
    Reply from 172.16.20.3: bytes=56 Sequence=5 ttl=255 time=10 ms
  --- 172.16.20.3 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 10/14/20 ms
```

其余直连网段的连通性测试省略。

2. 部署单区域 OSPF 网络

首先使用 **ospf** 命令创建并运行 OSPF。

```
<R1>system-view
[R1]ospf 1
```

其中，1 代表的是进程号，如果没有写明进程号，则默认是 1。

接着使用 **area** 命令创建区域并进入 OSPF 区域视图，输入要创建的区域 ID。由于本实验为 OSPF 单区域配置，所以使用骨干区域，即区域 0 即可。

```
[R1-ospf-1]area 0
```

再使用 **network** 命令来指定运行 OSPF 协议的接口和接口所属的区域。本实验中 R1 上的 3 个物理接口都需要指定。配置中需注意，尽量精确匹配所通告的网段。

```
[R1-ospf-1-area-0.0.0.0]network 172.16.10.0 0.0.0.255
[R1-ospf-1-area-0.0.0.0]network 172.16.20.0 0.0.0.255
[R1-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255
```

配置完成后使用 **display ospf interface** 命令检查 OSPF 接口通告是否正确。

```
[R1]display ospf interface
          OSPF Process 1 with Router-ID172.16.1.254
                  Interfaces
 Area: 0.0.0.0             (MPLS TE not enabled)
 IP Address       Type        State     Cost     Pri    DR              BDR
172.16.1.254     Broadcast   DR        1        1      172.16.1.254    0.0.0.0
172.16.10.1      Broadcast   DR        1        1      172.16.10.1     0.0.0.0
172.16.20.1      Broadcast   DR        1        1      172.16.20.1     0.0.0.0
```

可以观察到本地 OSPF 进程使用的 Router-ID 是 172.16.1.254。在此进程下，有 3 个接口加入了 OSPF 进程。“Type”为以太网默认的广播网络类型；“State”为该接口当前的状态，显示为 DR 状态，即表示为这 3 个接口在它们所在的网段中都被选举为 DR。

接下来在 R2 和 R3 上做相应配置，配置方法和 R1 相同，不再赘述。

```
[R2]ospf 1
[R2-ospf-1]area 0
[R2-ospf-1-area-0.0.0.0]network 172.16.10.0 0.0.0.255
```

```
[R2-ospf-1-area-0.0.0.0]network 172.16.30.0 0.0.0.255
[R2-ospf-1-area-0.0.0.0]network 172.16.2.0 0.0.0.255

[R3]ospf 1
[R3-ospf-1]area 0
[R3-ospf-1-area-0.0.0.0]network 172.16.20.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 172.16.30.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 172.16.3.0 0.0.0.255
```

3. 检查 OSPF 单区域的配置结果

以 R1 为例使用 **display ospf peer** 命令查看 OSPF 邻居状态。

```
<R1>display ospf peer
          OSPF Process 1 with Router-ID 172.16.1.254
                  Neighbors
 Area 0.0.0.0 interface 172.16.10.1(GigabitEthernet0/0/0)'s neighbors
Router-ID: 172.16.2.254 Address: 172.16.10.2
State: Full    Mode:Nbr is    Master    Priority: 1
   DR: 172.16.10.1    BDR: 172.16.10.2    MTU: 0
   Dead timer due in 35    sec
Retrans timer interval: 5
Neighbor is up for 00:07:38
 Authentication Sequence: [ 0 ]
                  Neighbors
 Area 0.0.0.0 interface 172.16.20.1(GigabitEthernet0/0/1)'s neighbors
Router-ID: 172.16.3.254 Address: 172.16.20.3
State: Full    Mode:Nbr is    Master    Priority: 1
   DR: 172.16.20.1    BDR: 172.16.20.3    MTU: 0
   Dead timer due in 29    sec
   Retrans timer interval: 5
Neighbor is up for 00:04:14
   Authentication Sequence: [ 0 ]
```

通过这条命令，可以查看很多内容。例如，通过 Router-ID 可以查看邻居的路由器标识；通过 Address 可以查看邻居的 OSPF 接口 IP 地址；通过 State 可以查看目前与该路由器的 OSPF 邻居状态；通过 Priority 可以查看当前该邻居 OSPF 接口的 DR 优先级等。

使用 **display ip routing-table protocol ospf** 命令查看 R1 上的 OSPF 路由表。

```
<R1>display ip routing-table protocol ospf
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Public routing table : OSPF
Destinations : 3          Routes : 4
OSPF routing table status : <Active>
Destinations : 3          Routes : 4
Destination/Mask     Proto    Pre   Cost      Flags   NextHop          Interface
172.16.2.0/24        OSPF     10    2          D     172.16.10.2     GigabitEthernet0/0/0
172.16.3.0/24        OSPF     10    2          D     172.16.20.3     GigabitEthernet0/0/1
172.16.30.0/24       OSPF     10    2          D     172.16.10.2     GigabitEthernet0/0/0
                     OSPF     10    2          D     172.16.20.3     GigabitEthernet0/0/1
OSPF routing table status : <Inactive>
Destinations : 0          Routes : 0
```

通过此路由表可以观察到，“Destination/Mask”标识了目的网段的前缀及掩码，“Proto”标识了此路由信息是通过 OSPF 协议获取的，“Pre”标识了路由优先级，“Cost”标识了开销值，“NextHop”标识了下一跳地址，“Interface”标识了此前缀

的出接口。

此时 R1 的路由表中已经拥有了去往网络中所有其他网段的路由条目。用同样方法查看 R2 与 R3 的 OSPF 邻居状态。

在 PC-1 上使用 **ping** 命令测试与 PC-3 间的连通性。

PC>ping 172.16.3.1

```
Ping 172.16.3.1: 32 data bytes, Press Ctrl_C to break
From 172.16.3.1: bytes=32 seq=1 ttl=126 time=31 ms
From 172.16.3.1: bytes=32 seq=2 ttl=126 time=32 ms
From 172.16.3.1: bytes=32 seq=3 ttl=126 time=15 ms
From 172.16.3.1: bytes=32 seq=4 ttl=126 time=16 ms
From 172.16.3.1: bytes=32 seq=5 ttl=126 time=16 ms
--- 172.16.3.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 15/22/32 ms
```

通信正常，其他测试省略。

思考

请列举链路状态协议与距离矢量路由协议的相同点与不同点。

8.2 OSPF 多区域配置

原理概述

在 OSPF 单区域中，每台路由器都需要收集其他所有路由器的链路状态信息，如果网络规模不断扩大，链路状态信息也会随之不断增多，这将使得单台路由器上链路状态数据库非常庞大，导致路由器负担加重，也不便于维护管理。为了解决上述问题，OSPF 协议可以将整个自治系统划分为不同的区域（Area），就像一个国家的国土面积很大时，会把整个国家划分为不同的省份来管理一样。

链路状态信息只在区域内部泛洪，区域之间传递的只是路由条目而非链路状态信息，因此大大减小了路由器的负担。当一台路由器属于不同区域时称它为区域边界路由器（Area Border Router，ABR），负责传递区域间路由信息。区域间的路由信息传递类似距离矢量算法，为了防止区域间产生环路，所有非骨干区域之间的路由信息必须经过骨干区域，也就是说非骨干区域必须和骨干区域相连，且非骨干区域之间不能直接进行路由信息交互。

实验目的

- 理解配置 OSPF 多区域的使用场景
- 掌握配置 OSPF 多区域的方法
- 理解 OSPF 区域边界路由器（ABR）的工作特点

实验内容

本实验模拟企业网络场景。R1、R2、R3、R4 为企业总部核心区域设备，属于区域 0，R5 属于新增分支机构 A 的网关设备，R6 属于新增分支机构 B 的网关设备。PC-1 和 PC-2 分别属于分支机构 A 和 B，PC-3 和 PC-4 属于总部管理员登录设备，用于管理网络。

在该网络中，如果设计方案采用单区域配置，则会导致单一区域 LSA 数目过于庞大，导致路由器开销过高，SPF 算法运算过于频繁。因此网络管理员选择配置多区域方案进行网络配置，将两个新分支运行在不同的 OSPF 区域中，其中 R5 属于区域 1，R6 属于区域 2。

实验拓扑

OSPF 多区域配置拓扑如图 8-2 所示。

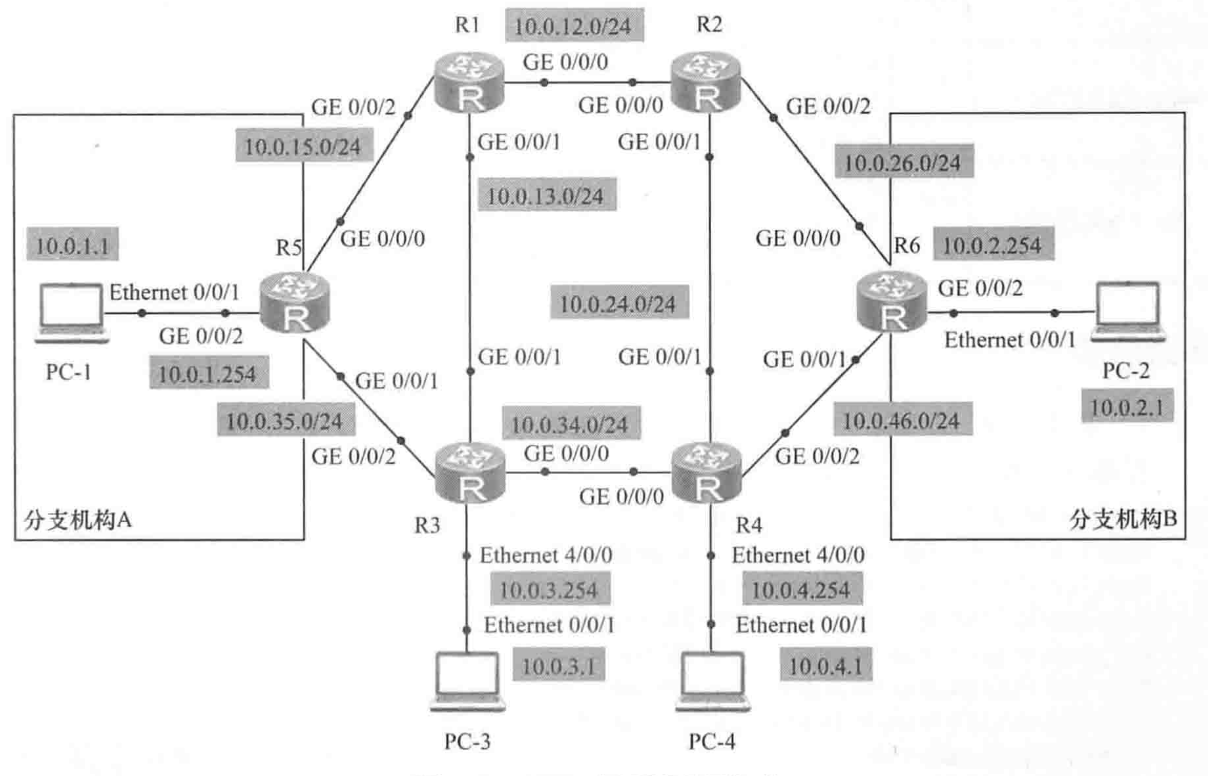


图 8-2 OSPF 多区域配置拓扑

实验编址

实验编址见表 8-2。

表 8-2 实验编址

设备	接口	IP 地址	子网掩码	默认网关
PC-1	Ethernet 0/0/1	10.0.1.1	255.255.255.0	10.0.1.254
PC-2	Ethernet 0/0/1	10.0.2.1	255.255.255.0	10.0.2.254
PC-3	Ethernet 0/0/1	10.0.3.1	255.255.255.0	10.0.3.254
PC-4	Ethernet 0/0/1	10.0.4.1	255.255.255.0	10.0.4.254

（续表）

设备	接口	IP 地址	子网掩码	默认网关
R1（AR2240）	GE 0/0/0	10.0.12.1	255.255.255.0	N/A
	GE 0/0/1	10.0.13.1	255.255.255.0	N/A
	GE 0/0/2	10.0.15.1	255.255.255.0	N/A
R2（AR2240）	GE 0/0/0	10.0.12.2	255.255.255.0	N/A
	GE 0/0/1	10.0.24.2	255.255.255.0	N/A
	GE 0/0/2	10.0.26.2	255.255.255.0	N/A
R3（AR2240）	GE 0/0/0	10.0.34.3	255.255.255.0	N/A
	GE 0/0/1	10.0.13.3	255.255.255.0	N/A
	GE 0/0/2	10.0.35.3	255.255.255.0	N/A
	Ethernet 4/0/0	10.0.3.254	255.255.255.0	N/A
R4（AR2240）	GE 0/0/0	10.0.34.4	255.255.255.0	N/A
	GE 0/0/1	10.0.24.4	255.255.255.0	N/A
	GE 0/0/2	10.0.46.4	255.255.255.0	N/A
	Ethernet 4/0/0	10.0.4.254	255.255.255.0	N/A
R5（AR2240）	GE 0/0/0	10.0.15.5	255.255.255.0	N/A
	GE 0/0/1	10.0.35.5	255.255.255.0	N/A
	GE 0/0/2	10.0.1.254	255.255.255.0	N/A
R6（AR2240）	GE 0/0/0	10.0.26.6	255.255.255.0	N/A
	GE 0/0/1	10.0.46.6	255.255.255.0	N/A
	GE 0/0/2	10.0.2.254	255.255.255.0	N/A

实验步骤

1. 基本配置

根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测各直连链路的连通性。

```
<R1>ping 10.0.15.5
PING 10.0.15.5: 56    data bytes, press CTRL_C to break
Reply from 10.0.15.5: bytes=56 Sequence=1 ttl=255 time=210 ms
Reply from 10.0.15.5: bytes=56 Sequence=2 ttl=255 time=60 ms
Reply from 10.0.15.5: bytes=56 Sequence=3 ttl=255 time=60 ms
Reply from 10.0.15.5: bytes=56 Sequence=4 ttl=255 time=30 ms
Reply from 10.0.15.5: bytes=56 Sequence=5 ttl=255 time=30 ms
--- 10.0.15.5 ping statistics ---
5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 30/78/210 ms
```

测试通过，其余直连网段的连通性测试省略。

2. 配置骨干区域路由器

在公司总部路由器 R1、R2、R3、R4 上创建 OSPF 进程，并在骨干区域 0 视图下通告总部各网段。

```
[R1]ospf 1
[R1-ospf-1]area 0
[R1-ospf-1-area-0.0.0.0]network 10.0.12.0 0.0.0.255
[R1-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255
```

```
[R2]ospf 1
[R2-ospf-1]area 0
[R2-ospf-1-area-0.0.0.0]network 10.0.12.0 0.0.0.255
[R2-ospf-1-area-0.0.0.0]network 10.0.24.0 0.0.0.255

[R3]ospf 1
[R3-ospf-1]area 0
[R3-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 10.0.34.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 10.0.3.0 0.0.0.255

[R4]ospf 1
[R4-ospf-1]area 0
[R4-ospf-1-area-0.0.0.0]network 10.0.34.0 0.0.0.255
[R4-ospf-1-area-0.0.0.0]network 10.0.24.0 0.0.0.255
[R4-ospf-1-area-0.0.0.0]network 10.0.4.0 0.0.0.255
```

配置完成后，测试总部内两台 PC 间的连通性。

```
PC>ping 10.0.4.1
Ping 10.0.4.1: 32 data bytes, Press Ctrl_C to break
From 10.0.4.1: bytes=32 seq=1 ttl=124 time=15 ms
From 10.0.4.1: bytes=32 seq=2 ttl=124 time=16 ms
From 10.0.4.1: bytes=32 seq=3 ttl=124 time=31 ms
From 10.0.4.1: bytes=32 seq=4 ttl=124 time=15 ms
From 10.0.4.1: bytes=32 seq=5 ttl=124 time=32 ms
--- 10.0.4.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 15/21/32 ms
```

已经可以正常通信，骨干区域路由器配置完成。

3. 配置非骨干区域路由器

在分支 A 的路由器 R5 上创建 OSPF 进程，创建并进入区域 1，并通告分支 A 的相应网段。

```
[R5]ospf 1
[R5-ospf-1]area 1
[R5-ospf-1-area-0.0.0.1]network 10.0.15.0 0.0.0.255
[R5-ospf-1-area-0.0.0.1]network 10.0.35.0 0.0.0.255
[R5-ospf-1-area-0.0.0.1]network 10.0.1.0 0.0.0.255
```

在 R1 和 R3 上也创建并进入区域 1 视图，将与 R5 相连的接口进行通告。

```
[R1]ospf 1
[R1-ospf-1]area 1
[R1-ospf-1-area-0.0.0.1]network 10.0.15.0 0.0.0.255

[R3]ospf 1
[R3-ospf-1]area 1
[R3-ospf-1-area-0.0.0.1]network 10.0.35.0 0.0.0.255
```

配置完成后，查看 OSPF 邻居状态。

```
[R5]display ospf peer
OSPF Process 1 with Router-ID 10.0.15.5
Neighbors
Area 0.0.0.1 interface 10.0.15.5(GigabitEthernet0/0/0)'s neighbors
Router-ID: 10.0.12.1              Address: 10.0.15.1
```

```
State: Full    Mode:Nbr is    Slave    Priority: 1
DR: 10.0.15.5   BDR: 10.0.15.1   MTU: 0
Dead timer due in 34    s
Retrans timer interval: 5
Neighbor is up for 00:07:46
Authentication Sequence: [ 0 ]
Neighbors
Area 0.0.0.1 interface 10.0.35.5(GigabitEthernet0/0/1)'s neighbors
Router-ID: 10.0.34.3              Address: 10.0.35.3
State: Full    Mode:Nbr is    Master    Priority: 1
DR: 10.0.35.5   BDR: 10.0.35.3   MTU: 0
Dead timer due in 36    s
Retrans timer interval: 5
Neighbor is up for 00:05:33
Authentication Sequence: [ 0 ]
```

可以观察到，现在 R5 与 R1 和 R3 的 OSPF 邻居关系建立正常，都为 Full 状态。

使用 **display ip routing-table protocol ospf** 命令查看 R5 路由表中的 OSPF 路由条目。

```
[R5]display ip routing-table protocol ospf
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Public routing table : OSPF
Destinations : 6          Routes : 8
OSPF routing table status : <Active>
Destinations : 6          Routes : 8
Destination/Mask    Proto   Pre    Cost    Flags    NextHop        Interface
10.0.3.0/24         OSPF    10     2       D        10.0.35.3      GigabitEthernet0/0/1
10.0.4.0/24         OSPF    10     3       D        10.0.35.3      GigabitEthernet0/0/1
10.0.12.0/24        OSPF    10     2       D        10.0.15.1      GigabitEthernet0/0/0
10.0.13.0/24        OSPF    10     2       D        10.0.15.1      GigabitEthernet0/0/0
                    OSPF    10     2       D        10.0.35.3      GigabitEthernet0/0/1
10.0.24.0/24        OSPF    10     3       D        10.0.15.1      GigabitEthernet0/0/0
                    OSPF    10     3       D        10.0.35.3      GigabitEthernet0/0/1
10.0.34.0/24        OSPF    10     2       D        10.0.35.3      GigabitEthernet0/0/1
OSPF routing table status : <Inactive>
Destinations : 0          Routes : 0
```

可以观察到，除 OSPF 区域 2 内的路由外，相关 OSPF 路由条目都已经获得。在拓扑中，R1 和 R3 这两台连接不同区域的路由器称为 ABR，即区域边界路由器，该类路由器设备可以同时属于两个以上的区域，但其中至少一个端口必须在骨干区域内。ABR 是用来连接骨干区域和非骨干区域的，其与骨干区域之间既可以是物理连接，也可以是逻辑上的连接。

使用 **display ospf lsdb** 命令查看 R5 的 OSPF 链路状态数据库信息。

```
<R5>display ospf lsdb
OSPF Process 1 with Router-ID 10.0.15.5
Link State Database
Area: 0.0.0.1
Type       LinkState ID    AdvRouter       Age   Len   Sequence    Metric
Router     10.0.12.1       10.0.12.1       1151  36    80000007    1
Router     10.0.34.3       10.0.34.3       1140  36    80000007    1
Router     10.0.15.5       10.0.15.5       1124  60    8000000C    1
Network    10.0.35.3       10.0.34.3       1140  32    80000004    0
Network    10.0.15.5       10.0.15.5       1145  32    80000005    0
Sum-Net    10.0.34.0       10.0.34.3       1137  28    80000005    1
```

```
Sum-Net    10.0.34.0    10.0.12.1     595  28   80000001    2
Sum-Net    10.0.13.0    10.0.12.1    1168  28   80000004    1
Sum-Net    10.0.13.0    10.0.34.3    1181  28   80000004    1
Sum-Net    10.0.12.0    10.0.12.1    1180  28   80000004    1
Sum-Net    10.0.12.0    10.0.34.3     516  28   80000001    2
Sum-Net    10.0.3.0     10.0.34.3    1179  28   80000004    1
Sum-Net    10.0.3.0     10.0.12.1    1125  28   80000004    2
```

可以观察到，关于其他区域的路由条目都是通过“Sum-Net”这类LSA获得，而这类LSA是不参与本区域的SPF算法运算的。

对公司另一分部B的路由器R6，和相应ABR设备R2、R4也做同样的配置。

```
[R6]ospf 1
[R6-ospf-1]area 2
[R6-ospf-1-area-0.0.0.2]network 10.0.26.0 0.0.0.255
[R6-ospf-1-area-0.0.0.2]network 10.0.46.0 0.0.0.255
[R6-ospf-1-area-0.0.0.2]network 10.0.2.0 0.0.0.255

[R2]ospf 1
[R2-ospf-1]area 2
[R2-ospf-1-area-0.0.0.2]network 10.0.26.0 0.0.0.255

[R4]ospf 1
[R4-ospf-1]area 2
[R4-ospf-1-area-0.0.0.2]network 10.0.46.0 0.0.0.255
```

配置完成，查看R6 的OSPF路由条目。

```
<R6>display ip routing-table protocol ospf
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Public routing table : OSPF
Destinations : 9          Routes : 12
OSPF routing table status : <Active>
Destinations : 8          Routes : 11
Destination/Mask   Proto   Pre   Cost   Flags   NextHop      Interface
10.0.1.0/24        OSPF    10    4      D       10.0.46.4    GigabitEthernet0/0/1
                   OSPF    10    4      D       10.0.26.2    GigabitEthernet0/0/0
10.0.3.0/24        OSPF    10    3      D       10.0.46.4    GigabitEthernet0/0/1
10.0.12.0/24       OSPF    10    2      D       10.0.26.2    GigabitEthernet0/0/0
10.0.13.0/24       OSPF    10    3      D       10.0.46.4    GigabitEthernet0/0/1
                   OSPF    10    3      D       10.0.26.2    GigabitEthernet0/0/0
10.0.15.0/24       OSPF    10    3      D       10.0.26.2    GigabitEthernet0/0/0
10.0.24.0/24       OSPF    10    2      D       10.0.26.2    GigabitEthernet0/0/0
                   OSPF    10    2      D       10.0.46.4    GigabitEthernet0/0/1
10.0.34.0/24       OSPF    10    2      D       10.0.46.4    GigabitEthernet0/0/1
10.0.35.0/24       OSPF    10    3      D       10.0.46.4    GigabitEthernet0/0/1
OSPF routing table status : <Inactive>
Destinations : 1          Routes : 1
```

观察到可以正常接收到所有OSPF路由信息。

测试分支A和分支B的PC-1和PC-2连通性。

```
PC>ping 10.0.2.1
Ping 10.0.2.1: 32 data bytes, Press Ctrl_C to break
From 10.0.2.1: bytes=32 seq=1 ttl=124 time=32 ms
From 10.0.2.1: bytes=32 seq=2 ttl=124 time=47 ms
From 10.0.2.1: bytes=32 seq=3 ttl=124 time=31 ms
```

```
From 10.0.2.1: bytes=32 seq=4 ttl=124 time=31 ms
From 10.0.2.1: bytes=32 seq=5 ttl=124 time=31 ms
--- 10.0.2.1 ping statistics ---
5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 31/34/47 ms
```

可以观察到，现在通信正常。至此，OSPF 多区域配置完成。

思考

在本实验中，如果现在公司总部配置的区域不是骨干区域 0，而是其他非骨干区域，会出现什么现象？

8.3 配置 OSPF 的认证

原理概述

OSPF 支持报文验证功能，只有通过验证的报文才能接收，否则将不能正常建立邻居关系。OSPF 协议支持两种认证方式——区域认证和链路认证。使用区域认证时，一个区域中所有的路由器在该区域下的认证模式和口令必须一致；OSPF 链路认证相比于区域认证更加灵活，可专门针对某个邻居设置单独的认证模式和密码。如果同时配置了接口认证和区域认证时，优先使用接口认证建立 OSPF 邻居。

每种认证方式又分为简单验证模式、MD5 验证模式和 Key chain 验证模式。简单验证模式在数据传递过程中，认证密钥和密钥 ID 都是明文传输，很容易被截获；MD5 验证模式下的密钥是经过 MD5 加密传输，相比于简单验证模式更为安全；Key chain 验证模式可以同时配置多个密钥，不同密钥可单独设置生效周期等。

实验目的

- 理解 OSPF 认证的应用场景
- 理解 OSPF 区域认证和链路认证的区别
- 掌握配置 OSPF 区域认证的方法
- 掌握配置 OSPF 链路认证的方法

实验内容

本实验模拟企业网络环境。R3、R5、R6 属于公司总部骨干区域路由器，R2 为 ABR。公司分部路由器 R1 和 R4 都属于区域 1，但分属不通部门，R1 作为市场部门网关，R4 作为财务部门网关。网络管理员在区域 0 和区域 1 上配置 OSPF 区域认证，其中区域 0 开启密文认证，区域 1 开启明文认证。为进一步提高该 OSPF 网络安全性，R2 和 R4 上单独设置密钥，配置 OSPF 链路认证。

实验拓扑

配置 OSPF 的认证拓扑如图 8-3 所示。

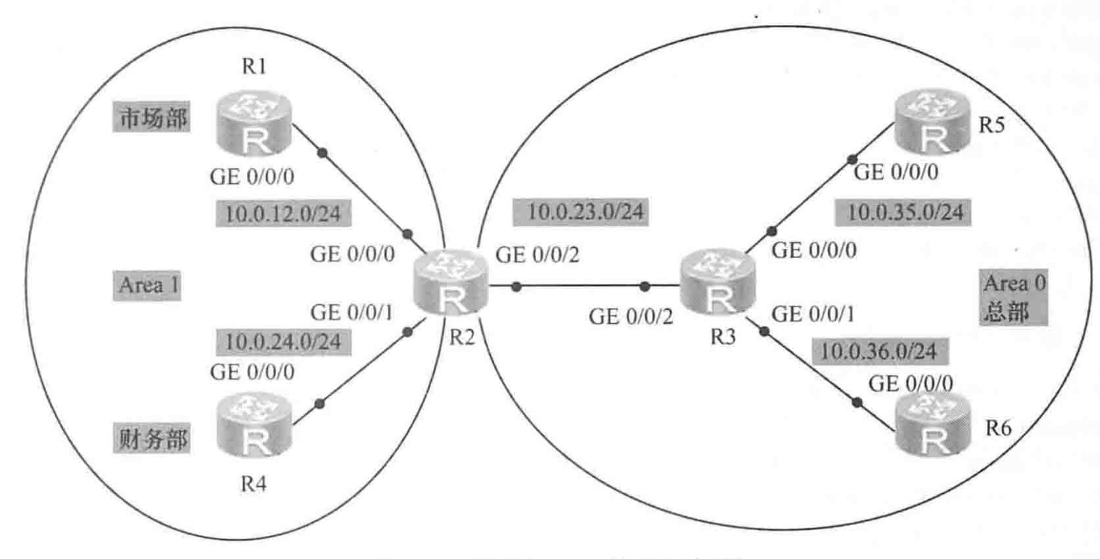


图 8-3 配置 OSPF 的认证拓扑

实验编址

实验编址见表 8-3。

表 8-3 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR2220）	Loopback0	1.1.1.1	255.255.255.255	N/A
	GE 0/0/0	10.0.12.1	255.255.255.0	N/A
R2（AR2220）	Loopback0	2.2.2.2	255.255.255.255	N/A
	GE 0/0/0	10.0.12.2	255.255.255.0	N/A
	GE 0/0/1	10.0.24.2	255.255.255.0	N/A
	GE 0/0/2	10.0.23.2	255.255.255.0	N/A
R3（AR2220）	Loopback0	3.3.3.3	255.255.255.255	N/A
	GE 0/0/0	10.0.35.3	255.255.255.0	N/A
	GE 0/0/1	10.0.36.3	255.255.255.0	N/A
	GE 0/0/2	10.0.23.3	255.255.255.0	N/A
R4（AR2220）	Loopback0	4.4.4.4	255.255.255.255	N/A
	GE 0/0/0	10.0.24.4	255.255.255.0	N/A
R5（AR2220）	Loopback0	5.5.5.5	255.255.255.255	N/A
	GE 0/0/0	10.0.35.5	255.255.255.0	N/A
R6（AR2220）	Loopback0	6.6.6.6	255.255.255.255	N/A
	GE 0/0/0	10.0.36.6	255.255.255.0	N/A

实验步骤

1. 基本配置

根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测各直连链路的连通性。

```
<R1>ping 10.0.12.2
PING 10.0.12.2: 56  data bytes, press CTRL_C to break
Reply from 10.0.12.2: bytes=56 Sequence=1 ttl=255 time=120 ms
Reply from 10.0.12.2: bytes=56 Sequence=2 ttl=255 time=90 ms
Reply from 10.0.12.2: bytes=56 Sequence=3 ttl=255 time=60 ms
Reply from 10.0.12.2: bytes=56 Sequence=4 ttl=255 time=40 ms
Reply from 10.0.12.2: bytes=56 Sequence=5 ttl=255 time=30 ms
--- 10.0.12.2 ping statistics ---
5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 30/68/120 ms
```

其余直连网段的连通性测试省略。

2. 搭建 OSPF 网络

在公司总部和分部各台路由器上进行相关 OSPF 多区域配置。

```
[R1]ospf 1
[R1-ospf-1]area 1
[R1-ospf-1-area-0.0.0.1]network 10.0.12.0 0.0.0.255
[R1-ospf-1-area-0.0.0.1]network 1.1.1.1 0.0.0.0

[R2]ospf 1
[R2-ospf-1]area 0
[R2-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255
[R2-ospf-1-area-0.0.0.0]network 2.2.2.2 0.0.0.0
[R2-ospf-1]area 1
[R2-ospf-1-area-0.0.0.1]network 10.0.12.0 0.0.0.255
[R2-ospf-1-area-0.0.0.1]network 10.0.24.0 0.0.0.255

[R3]ospf 1
[R3-ospf-1]area 0
[R3-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 10.0.35.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 10.0.36.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 3.3.3.3 0.0.0.0

[R4]ospf 1
[R4-ospf-1]area 1
[R4-ospf-1-area-0.0.0.1]network 10.0.24.0 0.0.0.255
[R4-ospf-1-area-0.0.0.1]network 4.4.4.4 0.0.0.0

[R5]ospf 1
[R5-ospf-1]area 0
[R5-ospf-1-area-0.0.0.0]network 10.0.35.0 0.0.0.255
[R5-ospf-1-area-0.0.0.0]network 5.5.5.5 0.0.0.0

[R6]ospf 1
[R6-ospf-1]area 0
[R6-ospf-1-area-0.0.0.0]network 10.0.36.0 0.0.0.255
[R6-ospf-1-area-0.0.0.0]network 6.6.6.6 0.0.0.0
```

其中每台设备上的环回口地址是为了后续实验中方便测试使用，所以需要通告到其所在区域。

配置完成后测试各设备上环回口的连通性。

```
<R1>ping 6.6.6.6
PING 6.6.6.6: 56  data bytes,  press CTRL_C to break
Reply from 6.6.6.6: bytes=56 Sequence=1 ttl=252 time=30 ms
Reply from 6.6.6.6: bytes=56 Sequence=2 ttl=252 time=30 ms
Reply from 6.6.6.6: bytes=56 Sequence=3 ttl=252 time=40 ms
Reply from 6.6.6.6: bytes=56 Sequence=4 ttl=252 time=30 ms
Reply from 6.6.6.6: bytes=56 Sequence=5 ttl=252 time=30 ms
--- 6.6.6.6 ping statistics ---
5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 30/32/40 ms
```

可以正常通信，其他环回口的测试省略。

3. 配置公司分部 OSPF 区域明文认证

网络管理员在公司分部的 OSPF 区域 1 中配置区域明文认证。

在 R1 上 OSPF 的区域 1 视图下使用 **authentication-mode** 命令指定该区域使用认证模式为 simple，即简单验证模式，配置口令为 huawei1，并配置 **plain** 参数。

配置 **plain** 参数后，可以使得在查看配置文件时，口令均以明文方式显示。如果不设置该参数的话，在查看配置文件时，默认会以密文方式显示口令，即无法查看到所配置的口令原文，这可以使非管理员用户在登录设备后无法查看到口令原文，从而提高安全性。

```
[R1]ospf 1
[R1-ospf-1]area 1
[R1-ospf-1-area-0.0.0.1]authentication-mode simple plain huawei1
[R1-ospf-1-area-0.0.0.1]display this
[V200R003C00]
#
 area 0.0.0.1
 authentication-mode simple plain huawei1
  network 1.1.1.1 0.0.0.0
  network 10.0.12.0 0.0.0.255
#
Return
```

可以观察到，此时以明文方式显示口令。

在 R1 上重新配置区域认证命令，并查看配置。

```
[R1-ospf-1-area-0.0.0.1]authentication-mode simple huawei1
[R1-ospf-1-area-0.0.0.1]display this
[V200R003C00]
#
 area 0.0.0.1
authentication-mode simple cipher %$%$P!/9Ac}_uERifpO}I^OJN<+@%$%$
network 1.1.1.1 0.0.0.0
  network 10.0.12.0 0.0.0.255
#
Return
```

可以观察到，默认情况下，口令以密文形式显示。

配置完成，等待 OSPF 网络收敛之后，查看 R1 与 R2 的 OSPF 邻居。

```
<R1>display ospf peer brief
```

```
            OSPF Process 1 with Router-ID 1
                  Peer Statistic Information
 ----------------------------------------------------------------------------
  Area Id          Interface                         Neighbor id      State
 ----------------------------------------------------------------------------
```

可以观察到，现在 R1 与 R2 邻居关系中断了，原因是目前仅仅在 R1 上配置了认证，导致 R1 和 R2 间的 OSPF 认证不匹配。

继续配置该区域的另一台设备 R2，必须要保证验证模式一致，口令也一致。

```
[R2]ospf 1
[R2-ospf-1]area 1
[R2-ospf-1-area-0.0.0.1]authentication-mode simple huawei1
```

配置完成后，等待一段时间，再次观察两者的邻居关系。

```
<R1>display ospf peer brief
            OSPF Process 1 with Router-ID 10.0.12.1
                  Peer Statistic Information
 ----------------------------------------------------------------------------
  Area Id          Interface                         Neighbor id      State
  0.0.0.1         GigabitEthernet0/0/0              10.0.12.2        Full
 ----------------------------------------------------------------------------
```

可以观察到，现在 R1 与 R2 的邻居关系状态恢复正常。

同理在 R4 上也做相同配置。

```
[R4]ospf 1
[R4-ospf-1]area 1
[R4-ospf-1-area-0.0.0.1]authentication-mode simple huawei1
```

配置完成后，在 R2 上查看 OSPF 邻居关系。

```
[R2]display ospf peer brief
            OSPF Process 1 with Router-ID 10.0.12.2
                  Peer Statistic Information
 ----------------------------------------------------------------------------
  Area Id          Interface                         Neighbor id      State
 0.0.0.0         GigabitEthernet0/0/2              10.0.23.3        Full
 0.0.0.1         GigabitEthernet0/0/0              10.0.12.1        Full
 0.0.0.1         GigabitEthernet0/0/1              10.0.24.4        Full
 ----------------------------------------------------------------------------
```

可以观察到，现在区域 1 中的邻居关系都建立正常。

4. 配置公司总部 OSPF 区域密文认证

根据设计，网络管理员在公司总部 OSPF 区域 0 中配置区域密文认证。

在 R2 上配置 OSPF Area 0 区域认证，使用验证模式为 **md5**，即 MD5 验证模式，验证字标识符为 1，配置口令为 huawei3。

```
[R2]ospf 1
[R2-ospf-1]area 0
[R2-ospf-1-area-0.0.0.0]authentication-mode md5 1 huawei3
```

继续在其他骨干路由器上做相同配置。注意，密文认证必须保证验证字标识符和口令完全一致认证才可以通过。

```
[R3]ospf 1
[R3-ospf-1]area 0
[R3-ospf-1-area-0.0.0.0]authentication-mode md5 1 huawei3
```

```
[R5]ospf 1
[R5-ospf-1]area 0
[R5-ospf-1-area-0.0.0.0]authentication-mode md5 1 huawei3

[R6]ospf 1
[R6-ospf-1]area 0
[R6-ospf-1-area-0.0.0.0]authentication-mode md5 1 huawei3
```

配置完成后，查看 R3 的 OSPF 邻居状态。

```
<R3>display ospf peer brief

         OSPF Process 1 with Router-ID 10.0.23.3
                  Peer Statistic Information
----------------------------------------------------------------------------
 Area Id          Interface                    Neighbor id        State
 0.0.0.0          GigabitEthernet0/0/2         10.0.12.2          Full
 0.0.0.0          GigabitEthernet0/0/0         10.0.35.5          Full
 0.0.0.0          GigabitEthernet0/0/1         10.0.36.6          Full
----------------------------------------------------------------------------
```

可以观察到，OSPF 邻居状态建立正常，其他设备上的查看过程省略。

5. 配置 OSPF 链路认证

在上面两个步骤中，使用了 OSPF 的区域认证方式配置了 OSPF 认证，使用链路认证方式配置可以达到同样的效果。如果采用链路认证的方式，就需要在同一 OSPF 的链路接口下都配置链路认证的命令，设置验证模式和口令等参数；而采用区域认证的方式时，在同一区域中，仅需在 OSPF 进程下的相应区域视图下配置一条命令来设置验证模式和口令即可，大大节省了配置量，所以在同一区域中如果有多台 OSPF 设备需要配置认证，建议选用区域认证的方式进行配置。

目前公司分部的 OSPF 区域中配置了简单模式的区域认证，为了进一步提升 R2 与 R4 之间的 OSPF 网络安全性，网络管理员需要在两台设备之间部署 MD5 验证模式的 OSPF 链路认证。

在 R2 的 GE 0/0/1 接口下使用 **ospf authentication-mode** 命令配置链路认证，配置使用 MD5 验证模式，验证字标识符为 1，口令为 huawei5。

```
[R2]interface GigabitEthernet 0/0/1
[R2-GigabitEthernet0/0/1]ospf authentication-mode md5 1 huawei5
```

配置完成后，等待一段时间，查看 R2 上的简要 OSPF 邻居信息。

```
[R2]display ospf peer brief
         OSPF Process 1 with Router-ID 10.0.12.2
                  Peer Statistic Information
----------------------------------------------------------------------------
 Area Id          Interface                    Neighbor id        State
 0.0.0.0          GigabitEthernet0/0/2         10.0.23.3          Full
 0.0.0.1          GigabitEthernet0/0/0         10.0.12.1          Full
----------------------------------------------------------------------------
```

发现 R2 与 R4 间的 OSPF 邻居关系已经消失。虽然已经配置好区域认证，但是如果同时配置了接口认证和区域认证时，会优先使用接口验证建立 OSPF 邻居。所以 R4 在没有配置链路认证之前，R2 与 R4 的邻居关系会因认证不匹配而无法建立。

同样在 R4 上配置链路，注意，验证模式、标识符、口令都需要保持一致。

```
[R4]interface GigabitEthernet 0/0/0
[R4-GigabitEthernet0/0/1]ospf authentication-mode md5 1 huawei5
```

配置完成后，等待一段时间，再次查看 R4 的 OSPF 邻居信息。

```
<R4>display ospf peer brief
         OSPF Process 1 with Router-ID 10.0.24.4
                  Peer Statistic Information
----------------------------------------------------------------------------
 Area Id          Interface                    Neighbor id       State
0.0.0.1         GigabitEthernet0/0/0          10.0.12.2         Full
----------------------------------------------------------------------------
```

可以观察到，邻居关系已经恢复正常。至此，OSPF 链路认证配置完成。

思考

OSPF 认证如果采用 MD5 验证模式，有没有办法可以获取其密钥内容？

8.4 OSPF 被动接口配置

原理概述

OSPF 被动接口也称抑制接口，成为被动接口后，将不会接收和发送 OSPF 报文。如果要使 OSPF 路由信息不被某一网络中的路由器获得且使本地路由器不接收网络中其他路由器发布的路由更新信息，即已运行在 OSPF 协议进程中的接口不与本链路上其余路由器建立邻居关系时，可通过配置被动接口来禁止此接口接收和发送 OSPF 报文。

实验目的

- 理解 OSPF 被动接口的应用场景
- 掌握 OSPF 被动接口的配置方法
- 理解 OSPF 被动接口的作用原理

实验内容

本实验模拟企业网络场景。有路由器 R1、R2、R4 与 R5 分属不同部门的网关设备，每台设备都连接着各部门的员工终端，公司整网运行 OSPF 协议，并都处于区域 0 中。员工终端上经常收到路由器发送的 OSPF 数据报文，而该报文对终端而言毫无用处，还占用了一定的链路带宽资源，并有可能引起安全风险，比如非法接入路由器做路由欺骗。现通告配置被动接口来实现阻隔 OSPF 报文，优化公司网络。

实验拓扑

OSPF 的被动接口拓扑如图 8-4 所示。

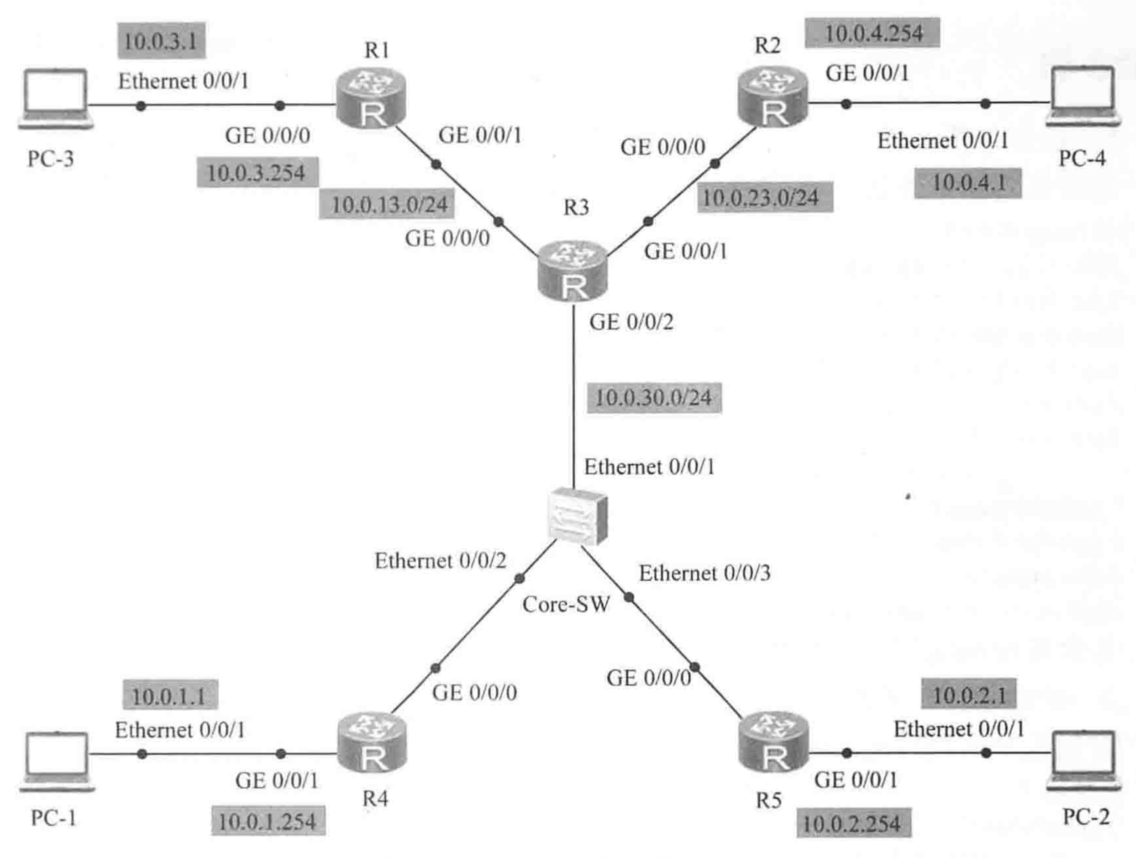


图 8-4 OSPF 的被动接口拓扑

实验编址

实验编址见表 8-4。

表 8-4 实验编址

设备	接口	IP 地址	子网掩码	默认网关
PC-1	Ethernet 0/0/1	10.0.1.1	255.255.255.0	10.0.1.254
PC-2	Ethernet 0/0/1	10.0.2.1	255.255.255.0	10.0.2.254
PC-3	Ethernet 0/0/1	10.0.3.1	255.255.255.0	10.0.3.254
PC-4	Ethernet 0/0/1	10.0.4.1	255.255.255.0	10.0.4.254
R1（AR2220）	GE 0/0/0	10.0.3.254	255.255.255.0	N/A
	GE 0/0/1	10.0.13.1	255.255.255.0	N/A
R2（AR2220）	GE 0/0/0	10.0.23.2	255.255.255.0	N/A
	GE 0/0/1	10.0.4.254	255.255.255.0	N/A
R3（AR2220）	GE 0/0/0	10.0.13.3	255.255.255.0	N/A
	GE 0/0/1	10.0.23.3	255.255.255.0	N/A
	GE 0/0/2	10.0.30.3	255.255.255.0	N/A
R4（AR2220）	GE 0/0/0	10.0.30.4	255.255.255.0	N/A
	GE 0/0/1	10.0.1.254	255.255.255.0	N/A
R5（AR2220）	GE 0/0/0	10.0.30.5	255.255.255.0	N/A
	GE 0/0/1	10.0.2.254	255.255.255.0	N/A

实验步骤

1. 基本配置

根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测各直连链路的连通性。

```
<R1>ping 10.0.3.1
PING 10.0.3.1: 56  data bytes,  press CTRL_C to break
Reply from 10.0.3.1: bytes=56 Sequence=1 ttl=128 time=30 ms
Reply from 10.0.3.1: bytes=56 Sequence=2 ttl=128 time=10 ms
Reply from 10.0.3.1: bytes=56 Sequence=3 ttl=128 time=1 ms
Reply from 10.0.3.1: bytes=56 Sequence=4 ttl=128 time=1 ms
Reply from 10.0.3.1: bytes=56 Sequence=5 ttl=128 time=10 ms
--- 10.0.3.1 ping statistics ---
5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 1/10/30 ms
```

其余直连网段的连通性测试省略。

2. 搭建 OSPF 网络

配置基本的 OSPF，所有路由器的接口都运行在区域 0 内。

```
[R1]ospf 1
[R1-ospf-1]area 0
[R1-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255
[R1-ospf-1-area-0.0.0.0]network 10.0.3.0 0.0.0.255

[R2]ospf 1
[R2-ospf-1]area 0
[R2-ospf-1-area-0.0.0.0]network 10.0.4.0 0.0.0.255
[R2-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255

[R3]ospf 1
[R3-ospf-1]area 0
[R3-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 10.0.30.0 0.0.0.255

[R4]ospf 1
[R4-ospf-1]area 0
[R4-ospf-1-area-0.0.0.0]network 10.0.30.0 0.0.0.255
[R4-ospf-1-area-0.0.0.0]network 10.0.1.0 0.0.0.255

[R5]ospf 1
[R5-ospf-1]area 0
[R5-ospf-1-area-0.0.0.0]network 10.0.30.0 0.0.0.255
[R5-ospf-1-area-0.0.0.0]network 10.0.2.0 0.0.0.255
```

配置完成之后，在 PC 上测试各网段间的连通性，以 PC-3 与 PC-4 间的连通性为例。

```
PC>ping 10.0.4.1
Ping 10.0.4.1: 32 data bytes,  Press Ctrl_C to break
From 10.0.4.1: bytes=32 seq=1 ttl=125 time=16 ms
From 10.0.4.1: bytes=32 seq=2 ttl=125 time=16 ms
From 10.0.4.1: bytes=32 seq=3 ttl=125 time=15 ms
From 10.0.4.1: bytes=32 seq=4 ttl=125 time=32 ms
From 10.0.4.1: bytes=32 seq=5 ttl=125 time=15 ms
```

```
--- 10.0.4.1 ping statistics ---
5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 15/18/32 ms
```

可以观察到，通信正常建立，其他测试省略。

在 PC-1 的接口 E 0/0/1 上抓包，如图 8-5 所示。

```
13 54.234000  10.0.1.254    224.0.0.5    OSPF   Hello Packet
14 63.719000  10.0.1.254    224.0.0.5    OSPF   Hello Packet
15 73.219000  10.0.1.254    224.0.0.5    OSPF   Hello Packet
16 82.703000  10.0.1.254    224.0.0.5    OSPF   Hello Packet
17 92.203000  10.0.1.254    224.0.0.5    OSPF   Hello Packet
18 101.703000 10.0.1.254    224.0.0.5    OSPF   Hello Packet
19 111.187000 10.0.1.254    224.0.0.5    OSPF   Hello Packet
20 120.687000 10.0.1.254    224.0.0.5    OSPF   Hello Packet
21 130.187000 10.0.1.254    224.0.0.5    OSPF   Hello Packet
```

图 8-5　抓包观察

发现该 PC 所在部门的网关路由器 R4 在不停地向这条线路发出 OSPF 的 Hello 报文尝试发现邻居，而对于 PC 而言，该报文是毫无用处的，同时也是不安全的。在 OSPF 的 Hello 报文中含有很多 OSPF 网络的重要信息，如果被恶意截取，容易出现安全隐患。

3. 配置被动接口

现在网络管理员通过配置被动接口来优化连接终端的网络，使终端不再收到任何 OSPF 报文。

在 R4 的 OSPF 进程中，使用 **silent-interface** 命令禁止接口接收和发送 OSPF 报文。

```
[R4]ospf 1
[R4-ospf-1]silent-interface GigabitEthernet 0/0/1
```

配置完成后，再次观察抓包结果，如图 8-6 所示。

No.	Time	Source	Destination	Protocol	Info
4	27.422000	10.0.1.254	224.0.0.5	OSPF	Hello Packet
5	36.594000	10.0.1.254	224.0.0.5	OSPF	Hello Packet
6	37.797000	10.0.1.254	224.0.0.5	OSPF	Hello Packet
7	47.063000	10.0.1.254	224.0.0.5	OSPF	Hello Packet
8	117.797000	HuaweiTe_cf:43:01	Broadcast	ARP	Who has 10.0.
9	117.891000	HuaweiTe_03:5e:38	HuaweiTe_cf:43:01	ARP	10.0.1.254 i
10	117.891000	10.0.1.1	10.0.1.254	ICMP	Echo (ping)
11	118.000000	10.0.1.254	10.0.1.1	ICMP	Echo (ping)
12	119.000000	10.0.1.1	10.0.1.254	ICMP	Echo (ping)

图 8-6　抓包观察

发现 OSPF 的报文在 Time 为 47 时间之后已经不再周期性发送任何 OSPF 的 Hello 报文。

如果 R4 上有多个接口需要设置为被动接口，只有 GE 0/0/1 接口保持活动状态，可以通过以下命令简化配置。

```
[R4]ospf 1
[R4-ospf-1]silent-interface all
[R4-ospf-1]undo silent-interface GigabitEthernet 0/0/1
```

这两种方法都可以将 GE 0/0/1 接口设置为被动接口，区别在于第一种方法只是单独

对某一个接口进行被动操作；而第二种是在对所有接口配置为被动接口后，再排除不需要配置为被动接口的接口。

同样在其他部门的网关路由器上进行相应配置，使得所有部门的终端都不再收到无关的 OSPF 报文。

```
[R1]ospf 1
[R1-ospf-1]silent-interface GigabitEthernet 0/0/0

[R2]ospf 1
[R2-ospf-1]silent-interface GigabitEthernet 0/0/1

[R5]ospf 1
[R5-ospf-1]silent-interface GigabitEthernet 0/0/1
```

4. 验证被动接口

配置被动接口，该接口会禁止接收和发送 OSPF 报文，故假使在两台路由器间 OSPF 链路的接口上也做该配置，会导致 OSPF 邻居的无法建立。

以 R5 为例，将其 GE 0/0/0 接口配置为被动接口。

```
[R5]ospf 1
[R5-ospf-1]silent-interface GigabitEthernet 0/0/0
```

配置完成后，查看 R5 的 OSPF 邻居关系状态。

```
<R5>display ospf peer
OSPF Process 1 with Router-ID 10.0.30.5
```

可以观察到，此时 R5 的 OSPF 邻居全部消失。

查看 R5 上的 OSPF 路由条目。

```
<R5>display ip routing-table protocol ospf
```

可以观察到，所有的 OSPF 路由条目都丢失。即验证了配置了被动接口后，OSPF 报文不再转发，包括建立邻居和维护邻居的 Hello 报文。

在上一步骤中，R4 的 GE 0/0/1 接口已经被配置了被动接口，那么配置该被动接口上的相关网段的路由信息能否正常地被其他邻居路由器收到？

在 R1 上查看 R4 被动接口 GE 0/0/1 上所连网段的路由条目 10.0.1.0/24。

```
<R1>display ip routing-table 10.0.1.1
Route Flags: R - relay,    D - download to fib
------------------------------------------------------------------------------
Routing Table : Public
Summary Count : 1
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
10.0.1.0/24         OSPF    10      3        D   10.0.13.3   GigabitEthernet0/0/1
```

可以观察到，此时其他邻居路由器仍然可以收到该网段的路由条目。

被动接口特性为只是不再收发任何 OSPF 协议报文，但是被动接口所在网段的直连路由条目如果已经在 OSPF 中通告，那么也会被其他的 OSPF 邻居路由器接收到。

在 PC-1 上，测试与 PC-4 之间的连通性。

```
PC>ping 10.0.4.1
Ping 10.0.4.1: 32 data bytes, Press Ctrl_C to break
From 10.0.4.1: bytes=32 seq=1 ttl=125 time=32 ms
From 10.0.4.1: bytes=32 seq=2 ttl=125 time=46 ms
From 10.0.4.1: bytes=32 seq=3 ttl=125 time=47 ms
From 10.0.4.1: bytes=32 seq=4 ttl=125 time=32 ms
From 10.0.4.1: bytes=32 seq=5 ttl=125 time=31 ms
```

```
--- 10.0.4.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 31/37/47 ms
```

可以观察到，通信正常，完全不受影响。

思考

在本实验中，通过配置被动接口可以禁止 OSPF 收发 Hello 报文，是否还有其他办法也能实现？

8.5 理解 OSPF Router-ID

原理概述

一些动态路由协议要求使用 Router-ID 作为路由器的身份标示，如果在启动这些路由协议时没有指定 Router-ID，则默认使用路由器全局下的路由管理 Router-ID。

Router-ID 选举规则为，如果通过 **Router-ID** 命令配置了 Router-ID，则按照配置结果设置。在没有配置 Router-ID 的情况下，如果存在配置了 IP 地址的 Loopback 接口，则选择 Loopback 接口地址中最大的地址作为 Router-ID；如果没有已配置 IP 地址的 Loopback 接口，则从其他接口的 IP 地址中选择最大的地址作为 Router-ID（不考虑接口的 Up/Down 状态）。

当且仅当被选为 Router-ID 的接口 IP 地址被删除/修改，才触发重新选择过程，其他情况（例如接口处于 DOWN 状态；已经选取了一个非 Loopback 接口地址后又配置了一个 Loopback 接口地址；配置了一个更大的接口地址等）不触发重新选择的过程。

Router-ID 改变之后，各协议需要通过手工执行 **reset** 命令才会重新选取新的 Router-ID。

实验目的

- 理解 Router-ID 的选举规则
- 掌握 OSPF 手动配置 Router-ID 的方法
- 理解 OSPF 中 Router-ID 必须唯一的意义

实验内容

本实验模拟企业网络环境。R1 为部门 A 的网关设备，R3 为部门 B 的网关设备，R4 为部门 C 的网关设备，R2 为企业核心路由器。现网络中运行 OSPF 协议实现全网互通，所有路由器运行在区域 0 内，网络管理员需要正确配置 Router-ID 以避免产生不必要的问题。

实验拓扑

理解 OSPF 的 Router-ID 拓扑如图 8-7 所示。

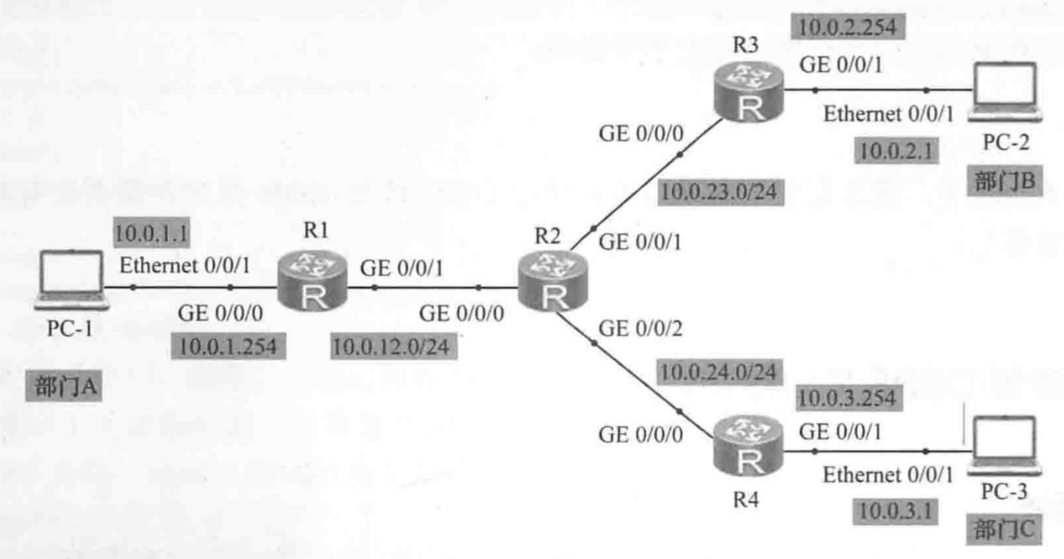


图 8-7 理解 OSPF 的 Router-ID 拓扑

实验编址

实验编址见表 8-5。

表 8-5 实验编址

设备	接口	IP 地址	子网掩码	默认网关
PC-1	Ethernet 0/0/1	10.0.1.1	255.255.255.0	10.0.1.254
PC-2	Ethernet 0/0/1	10.0.2.1	255.255.255.0	10.0.2.254
PC-3	Ethernet 0/0/1	10.0.3.1	255.255.255.0	10.0.3.254
R1（AR2220）	Loopback 0	1.1.1.1	255.255.255.255	N/A
	GE 0/0/0	10.0.1.254	255.255.255.0	N/A
	GE 0/0/1	10.0.12.1	255.255.255.0	N/A
R2（AR2220）	Loopback 0	2.2.2.2	255.255.255.255	N/A
	GE 0/0/0	10.0.12.2	255.255.255.0	N/A
	GE 0/0/1	10.0.23.2	255.255.255.0	N/A
	GE 0/0/2	10.0.24.2	255.255.255.0	N/A
R3（AR2220）	Loopback 0	3.3.3.3	255.255.255.255	N/A
	GE 0/0/0	10.0.23.3	255.255.255.0	N/A
	GE 0/0/1	10.0.2.254	255.255.255.0	N/A
R4（AR2220）	Loopback 0	4.4.4.4	255.255.255.255	N/A
	GE 0/0/0	10.0.24.4	255.255.255.0	N/A
	GE 0/0/1	10.0.3.254	255.255.255.0	N/A

实验步骤

1．验证 Router-ID 选举规则

在进行基本配置之前，在 R1 上使用 **display Route-ID** 命令来查看当前设备上的 Router-ID。

```
[R1]display Router-ID
Router-ID:0.0.0.0
```

可以观察到，在设备没有配置任何接口时，Router-ID 为 0.0.0.0。

根据实验编址表，在 R1 的 GE 0/0/1 接口上配置 IP 地址 10.0.12.1，GE 0/0/0 接口配置 IP 地址 10.0.1.254，配置环回接口 0 的地址 1.1.1.1。

```
[R1]interface gigabitethernet 0/0/1
[R1-GigabitEthernet0/0/1]ip addres 10.0.12.1 24
[R1-GigabitEthernet0/0/1]interface gigabitethernet 0/0/0
[R1-GigabitEthernet0/0/0]ip addres 10.0.1.254 24
[R1-GigabitEthernet0/0/0]interface loopback 0
[R1-LoopBack0]ip addres1.1.1.1 32
```

配置完成后，在 R1 上查看所有接口信息。

```
<R1>display ip interface brief
*down: administratively down
^down: standby
(l): loopback
(s): spoofing
The number of interface that is UP in Physical is 4
The number of interface that is DOWN in Physical is 1
The number of interface that is UP in Protocol is 4
The number of interface that is DOWN in Protocol is 1
Interface                     IP Address/Mask      Physical    Protocol
GigabitEthernet0/0/0          10.0.1.254/24        up          up
GigabitEthernet0/0/1          10.0.12.1/24         up          up
GigabitEthernet0/0/2          unassigned           down        down
LoopBack0                     1.1.1.1/32           up          up(s)
NULL0                         unassigned           up          up(s)
```

可以观察到，目前所配置的接口及 IP 地址信息。

查看当前设备上的 Router-ID。

```
[R1]display Router-ID
Router-ID:10.0.12.1
```

可以观察到当前设备上的全局 Router-ID 为 10.0.12.1，而不是环回接口地址 1.1.1.1，这是为什么？

原因是接口配置顺序会影响 Router-ID 的选举，因为设备上第一次配置的是物理接口的地址，该动作便会触发 Router-ID 的选举。而此刻，设备上也有且仅有该物理地址，所以该地址便会被 Router-ID 所使用，后续即使再配置了环回接口地址也不会使用。同理，如果第一次配置的是其他物理接口的地址，或者是环回接口的地址，都会被 Router-ID 所使用。

在 R1 上删除接口 GE 0/0/1 的 IP 地址，并再次查看此时设备的 Router-ID。

```
[R1]interface gigabitethernet 0/0/1
[R1-GigabitEthernet0/0/1]undo ip address

[R1]display Router-ID
Router-ID:1.1.1.1
```

可以观察到，当删除当前 Router-ID 所使用的 IP 地址时，便会触发重新选举，按照环回接口优先的规则选择使用 1.1.1.1 作为 Router-ID。

可以采用手动配置的方式强制指定 R1 的 Router-ID 为 1.1.1.1。这样配置的优点是，即使该地址现在已经不是 R1 的任何接口的地址，也可以修改成为 Router-ID（删除该环回接口也不会触发重新选举，验证省略）。

```
[R1]Router-ID 1.1.1.1
```

配置完成后，马上会弹出以下信息：

```
Info: Router-ID has been modified, please reset the relative protocols manually to update the Router-ID.
```

该信息表示 Router-ID 已经被修改，请重启相应的路由协议进行更新。即当前全局配置的 Router-ID 已经被更新，如果目前设备上已经运行了 OSPF 协议，需要重置 OSPF 协议进程或者重启整台路由器才可以使得 OSPF 协议中的 Router-ID 也同步更新使用该新的全局 Router-ID。需要使用 **reset ospf process** 命令来重置 OSPF 协议进程。

2. 基本配置

根据实验编址表进行完成剩余基本配置，并使用 **ping** 命令检测各直连链路的连通性。

```
<R1>ping 10.0.1.1
  PING 10.0.1.1: 56  data bytes, press CTRL_C to break
    Reply from 10.0.1.1: bytes=56 Sequence=1 ttl=128 time=170 ms
    Reply from 10.0.1.1: bytes=56 Sequence=2 ttl=128 time=70 ms
    Reply from 10.0.1.1: bytes=56 Sequence=3 ttl=128 time=30 ms
    Reply from 10.0.1.1: bytes=56 Sequence=4 ttl=128 time=30 ms
    Reply from 10.0.1.1: bytes=56 Sequence=5 ttl=128 time=10 ms
  --- 10.0.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 10/62/170 ms
```

其余直连网段的连通性测试省略。

3. 理解 OSPF 的 Router-ID

在所有路由器上配置 OSPF 协议，并都运行在区域 0 内。使用 **ospf router-id** 命令来配置 OSPF 协议的私有 Router-ID，如果不配置，则默认使用全局下的 Router-ID。

注意区分设备全局下的 Router-ID 和路由协议的 Router-ID 的概念。如果在路由协议中没有配置 Router-ID，就会默认使用路由器的全局 Router-ID。如果配置，则可以和全局 Router-ID 不一致。

一般建议采用环回接口地址作为路由协议的 Router-ID，因为环回接口是逻辑接口，比物理接口更加稳定。在对网络操作时，网络管理员有可能误操作导致物理接口地址删除，或者改动，而环回接口则一般不会去改动。

```
[R1]ospf 1 router-id 1.1.1.1
[R1-ospf-1]area 0
[R1-ospf-1-area-0.0.0.0]network 10.0.12.0 0.0.0.255
[R1-ospf-1-area-0.0.0.0]network 10.0.1.0 0.0.0.255

[R2]ospf 1 router-id 2.2.2.2
[R2-ospf-1]area 0
[R2-ospf-1-area-0.0.0.0]network 10.0.12.0 0.0.0.255
[R2-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255
[R2-ospf-1-area-0.0.0.0]network 10.0.24.0 0.0.0.255
```

```
[R3]ospf 1 router-id 3.3.3.3
[R3-ospf-1]area 0
[R3-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 10.0.2.0 0.0.0.255

[R4]ospf 1 router-id 4.4.4.4
[R4-ospf-1]area 0
[R4-ospf-1-area-0.0.0.0]network 10.0.24.0 0.0.0.255
[R4-ospf-1-area-0.0.0.0]network 10.0.3.0 0.0.0.255
```

配置完成后测试 PC-1 和 PC-2 间的连通性。

```
PC>ping 10.0.2.1
Ping 10.0.2.1: 32 data bytes, Press Ctrl_C to break
From 10.0.2.1: bytes=32 seq=1 ttl=124 time=31 ms
From 10.0.2.1: bytes=32 seq=2 ttl=124 time=47 ms
From 10.0.2.1: bytes=32 seq=3 ttl=124 time=31 ms
From 10.0.2.1: bytes=32 seq=4 ttl=124 time=32 ms
From 10.0.2.1: bytes=32 seq=5 ttl=124 time=31 ms
--- 10.0.2.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 31/34/47 ms
```

可以观察到，目前通信正常，其余 PC 间连通性测试省略。

现在修改 R2 的 Router-ID 为 3.3.3.3，即 R3 的 Router-ID，使 R3 和 R2 的 Router-ID 重叠，并重置协议进程使该配置生效。

```
<R2>ospf 1 router-id 3.3.3.3
<R2>reset ospf process
Warning: The OSPF process will be reset. Continue? [Y/N]:y
```

待协议收敛后，再次查看 R2 的 OSPF 邻居信息。

```
<R2>display ospf peer
         OSPF Process 1 with Router-ID 3.3.3.3
                 Neighbors
 Area 0.0.0.0 interface 10.0.12.2(GigabitEthernet0/0/0)'s neighbors
Router-ID: 1.1.1.1              Address: 10.0.12.1
   State: Full   Mode:Nbr is   Slave   Priority: 1
   DR: 10.0.12.1   BDR: 10.0.12.2   MTU: 0
   Dead timer due in 40   sec
   Retrans timer interval: 5
   Neighbor is up for 00:01:36
   Authentication Sequence: [ 0 ]
              Neighbors
 Area 0.0.0.0 interface 10.0.24.2(GigabitEthernet0/0/2)'s neighbors
 Router ID: 4.4.4.4              Address: 10.0.24.4
   State: Full   Mode:Nbr is   Master   Priority: 1
   DR: 10.0.24.4   BDR: 10.0.24.2   MTU: 0
   Dead timer due in 40   sec
   Retrans timer interval: 5
   Neighbor is up for 00:00:40
Authentication Sequence: [ 0 ]
```

可以观察到到 R2 与 R3 的邻居关系消失。

测试 PC-1 与 PC-2 的连通性。

```
PC>ping 10.0.2.1
```

```
Ping 10.0.2.1: 32 data bytes, Press Ctrl_C to break
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
--- 10.0.2.1 ping statistics ---
  5 packet(s) transmitted
  0 packet(s) received
  100.00% packet loss
```

网络已经发生故障，无法正常通信。验证了 OSPF 建立直连邻居关系时，Router-ID 一定不能重叠。那么如果 OSPF 非直连邻居的 Router-ID 重叠会产生什么现象？

还原 R2 之前的配置，调整 R4 的 Router-ID 为 3.3.3.3，与 R3 重叠。

```
<R2>ospf 1 router-id 2.2.2.2
<R2>reset ospf process
Warning: The OSPF process will be reset. Continue? [Y/N]:y

<R4>ospf 1 router-id 3.3.3.3
<R4>reset ospf process
Warning: The OSPF process will be reset. Continue? [Y/N]:y
```

配置完成后，查看 R2 的 OSPF 邻居状态。

```
<R2>display ospf peer brief
          OSPF Process 1 with Router-ID 2.2.2.2
                    Peer Statistic Information
---------------------------------------------------------------------------
Area Id          Interface                      Neighbor id       State
0.0.0.0          GigabitEthernet0/0/0           1.1.1.1           Full
0.0.0.0          GigabitEthernet0/0/1           3.3.3.3           Full
0.0.0.0          GigabitEthernet0/0/2           3.3.3.3           Full
---------------------------------------------------------------------------
```

发现 R2 有两个 3.3.3.3 的邻居，查看 R2 的路由表。

```
<R2>display ip routing-table protocol ospf
Route Flags: R - relay, D - download to fib
---------------------------------------------------------------------------
Public routing table : OSPF
Destinations : 2          Routes : 2
OSPF routing table status : <Active>
Destinations : 2          Routes : 2
Destination/Mask    Proto   Pre  Cost    Flags NextHop       Interface
10.0.1.0/24         OSPF    10   2       D    10.0.12.1    GigabitEthernet0/0/0
10.0.3.0/24         OSPF    10   2       D    10.0.24.4    GigabitEthernet0/0/2
OSPF routing table status : <Inactive>
          Destinations : 0          Routes : 0
```

可以观察到，此时 R2 没有接收到 R3 上 10.0.2.0/24 网段的路由条目，即使路由器邻居关系建立正常，但也无法正常获取路由条目。

测试 PC-1 与 PC-2 间的连通性。

```
PC>ping 10.0.2.1
Ping 10.0.2.1: 32 data bytes, Press Ctrl_C to break
Request timeout!
Request timeout!
Request timeout!
Request timeout!
```

```
Request timeout!
......
```

通信无法正常进行。这是因为 R2 认为是同一个 OSPF 邻居，但是 LSA 又不一致，造成链路状态数据库发送错误，无法计算出正确的路由信息。

综上所述，OSPF 协议的 Router-ID 务必要在整个路由选择域内保持唯一。

思考

试问如果不同区域中的 OSPF 路由器的 Router-ID 重叠又会导致什么问题的产生？

8.6 OSPF 的 DR 与 BDR

原理概述

在 OSPF 的广播类型网络和 NBMA 类型网络中，如果网络中有 n 台路由器，若任意两台路由器之间都要建立邻接关系，则需要建立 $n\times(n-1)/2$ 个邻接关系，即当路由器很多时，则需要建立和维护的邻接关系就很多，两两之间需要发送的报文也就很多，这会造成很多内容重复的报文在网络中传递，浪费了设备的带宽资源。因此在广播和 NBMA 类型网络中，OSPF 协议定义了指定路由器 DR（Designated Router），即所有其他路由器都只将各自的链路状态信息发送给 DR，再由 DR 以组播方式发送至所有路由器，大大减少了 OSPF 数据包的发送。

但是如果 DR 由于某种故障而失效，此时网络中必须重新选举 DR，并同步链路状态信息，这需要较长的时间。为了能够缩短这个过程，OSPF 协议又定义了 BDR（Backup Designated Router）的概念，作为 DR 路由器的备份，当 DR 路由器失效时，BDR 成为 DR，并再选择新的 BDR 路由器。其他非 DR/BDR 路由器都称为 DR Other 路由器。

每一个含有至少两个路由器的广播类型网络或NBMA类型网络都会选举一个DR和BDR。选举规则是首先比较 DR 优先级，优先级高者成为 DR，次高的成为 BDR。如果优先级相等，则 Router-ID 数值高的成为 DR，次高的成为 BDR。如果一台路由器的 DR 优先级为 0，则不参与选举。需要注意的是，DR 是在某个广播或者 NBMA 网段内进行选举的，是针对路由器的接口而言的。某台路由器在一个接口上可能是 DR，在另一个接口上有可能是 BDR，或者是 DR Other。

若 DR、BDR 已经选举完毕，人为修改任何一台路由器的 DR 优先级值为最大，也不会抢占成为新的 DR 或 BDR，即 OSPF 的 DR/BDR 选举是非抢占的。

实验目的

- 理解 OSPF 在哪种网络类型中会选举 DR/BDR
- 掌握 OSPF DR/BDR 的选举规则
- 掌握如何更改设备接口上的 DR 优先级
- 理解 OSPF DR/BDR 选举的非抢占特性

实验内容

某公司有 4 个部门，路由器 R1 连接到总经理办公室，路由器 R2 连接到人事部，R3 连接到开发部，R4 连接到市场部。4 台路由器通过交换机 S1 互联，每台路由器都运行了 OSPF 路由协议，都运行在区域 0 内，使得公司内部各部门网络能够互相通信。由于路由器通过广播网络互连，OSPF 会选举 DR 和 BDR，现网络管理员要配置使得性能较好的 R1 成为 DR，性能次之的 R2 成为 BDR，而性能最差的 R4 不能参加 DR 和 BDR 的选举，由此来完成网络的优化。

实验拓扑

本实验的拓扑如图 8-8 所示。

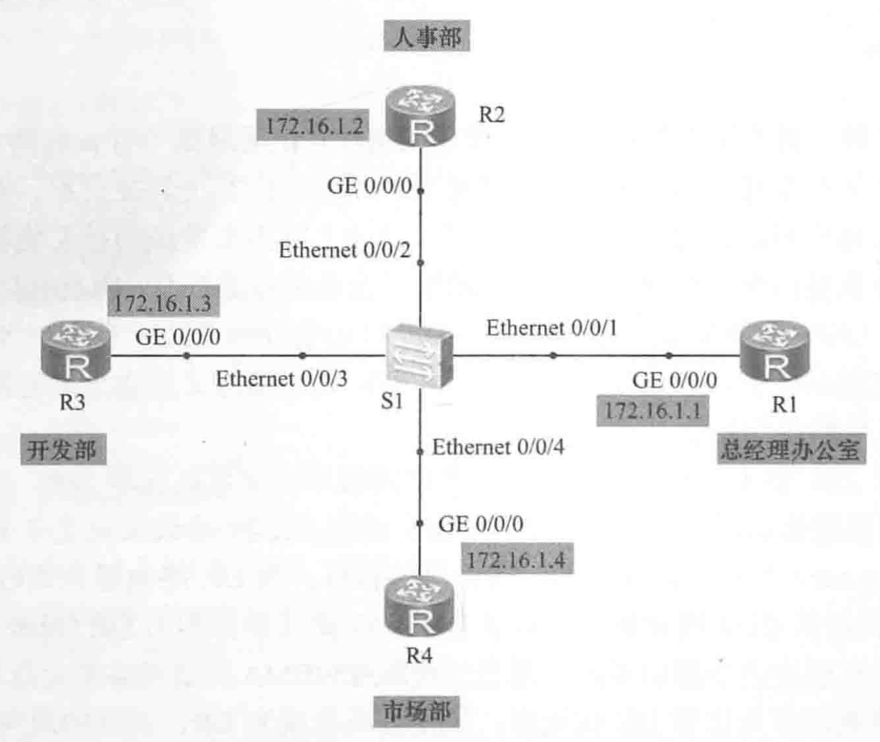


图 8-8 理解 OSPF 的 DR 与 BDR 拓扑

实验编址

实验编址见表 8-6。

表 8-6 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR2220）	GE 0/0/0	172.16.1.1	255.255.255.0	N/A
	Loopback 0	1.1.1.1	255.255.255.255	N/A
R2（AR2220）	GE 0/0/0	172.16.1.2	255.255.255.0	N/A
	Loopback 0	2.2.2.2	255.255.255.255	N/A
R3（AR2220）	GE 0/0/0	172.16.1.3	255.255.255.0	N/A
	Loopback 0	3.3.3.3	255.255.255255	N/A
R4（AR2220）	GE 0/0/0	172.16.1.4	255.255.255.0	N/A
	Loopback 0	4.4.4.4	255.255.255.255	N/A

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 地址配置，并使用 **ping** 命令检测各直连链路的连通性。

```
<R1>ping 172.16.1.4
  PING 172.16.1.4: 56  data bytes, press CTRL_C to break
    Reply from 172.16.1.4: bytes=56 Sequence=1 ttl=255 time=100 ms
    Reply from 172.16.1.4: bytes=56 Sequence=2 ttl=255 time=70 ms
    Reply from 172.16.1.4: bytes=56 Sequence=3 ttl=255 time=70 ms
    Reply from 172.16.1.4: bytes=56 Sequence=4 ttl=255 time=30 ms
    Reply from 172.16.1.4: bytes=56 Sequence=5 ttl=255 time=50 ms
  --- 172.16.1.4 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 30/64/100 ms
```

其余直连网段的连通性测试省略。

2. 搭建基本的 OSPF 网络

在公司网络中的 4 台路由器 R1、R2、R3、R4 上配置基础的 OSPF 网络配置。每台路由器使用各自的环回接口地址作为 Router-ID，并且都运行在区域 0 内。

```
[R1]router id 1.1.1.1
[R1]ospf 1
[R1-ospf-1]area 0
[R1-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255

[R2]router id 2.2.2.2
[R2]ospf 1
[R2-ospf-1]area 0
[R2-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255

[R3]router id 3.3.3.3
[R3]ospf 1
[R3-ospf-1]area 0
[R3-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255

[R4]router id 4.4.4.4
[R4]ospf 1
[R4-ospf-1]area 0
[R4-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255
```

配置完成后，同时重启 4 台路由器上的 OSPF 进程，或者直接同时重启设备。

```
<R1>reset ospf process

<R2>reset ospf process

<R3>reset ospf process

<R4>reset ospf process
```

重置后再次检查 OSPF 邻居建立情况，使用 **display ospf peer brief** 命令进行查看。

```
<R1>display ospf peer brief
```

```
          OSPF Process 1 with Router-ID 1.1.1.1
                  Peer Statistic Information
 ----------------------------------------------------------------------------
  Area Id        Interface                     Neighbor id     State
  0.0.0.0        GigabitEthernet0/0/0          2.2.2.2         Full
  0.0.0.0        GigabitEthernet0/0/0          3.3.3.3         Full
  0.0.0.0        GigabitEthernet0/0/0          4.4.4.4         Full
 ----------------------------------------------------------------------------
```

可以观察到，R1 此时已经和其他路由器成功建立起 OSPF 邻居关系。其他设备上的查看过程省略。

3. 查看默认情况下的 DR/BDR 状态

使用 **display ospf peer** 命令查看此时默认情况下 OSPF 网络中的 DR/BDR 选举情况。

```
[R1]display ospf peer
          OSPF Process 1 with Router-ID 1.1.1.1
                 Neighbors
Area 0.0.0.0 interface 172.16.1.1(GigabitEthernet0/0/0)'s neighbors
Router-ID: 2.2.2.2              Address: 172.16.1.2
State: 2-Way    Mode:Nbr is    Master    Priority: 1
DR: 172.16.1.4    BDR: 172.16.1.3    MTU: 0
Dead timer due in 35    s
Retrans timer interval: 0
Neighbor is up for 00:00:00
Authentication Sequence: [ 0 ]
Router-ID: 3.3.3.3              Address: 172.16.1.3
State: Full    Mode:Nbr is    Master    Priority: 1
DR: 172.16.1.4    BDR: 172.16.1.3    MTU: 0
Dead timer due in 34    s
Retrans timer interval: 5
Neighbor is up for 00:00:48
Authentication Sequence: [ 0 ]
Router-ID: 4.4.4.4              Address: 172.16.1.4
State: Full    Mode:Nbr is    Master    Priority: 1
DR: 172.16.1.4    BDR: 172.16.1.3    MTU: 0
Dead timer due in 34    s
Retrans timer interval: 0
Neighbor is up for 00:00:46
Authentication Sequence: [ 0 ]
```

可以观察到在该广播网络中，此时 R4 为 OSPF 网络中的 DR，R3 为 BDR。这是由于在默认情况下，每台路由器上的 DR 优先级都为 1，此时是通过 Router-ID 的数值高低进行比较的。

接下来在每台设备上的相关接口下使用 **ospf network-type p2mp** 命令修改 OSPF 的网络类型为点到多点。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]ospf network-type p2mp

[R2]interface GigabitEthernet 0/0/0
[R2-GigabitEthernet0/0/0]ospf network-type p2mp

[R3]interface GigabitEthernet 0/0/0
[R3-GigabitEthernet0/0/0]ospf network-type p2mp
```

```
[R4]interface GigabitEthernet 0/0/0
[R4-GigabitEthernet0/0/0]ospf network-type p2mp
```

配置完成后，在 R1 上再次观察此时 OSPF 的 DR/BDR 选举情况。

```
[R1]display ospf peer
          OSPF Process 1 with Router-ID 1.1.1.1
                   Neighbors
   Area 0.0.0.0 interface 172.16.1.1(GigabitEthernet0/0/0)'s neighbors
   Router-ID: 2.2.2.2            Address: 172.16.1.2
   State: Full   Mode:Nbr is   Master   Priority: 1
   DR: None     BDR: None     MTU: 0
   Dead timer due in 109 sec
   Retrans timer interval: 0
   Neighbor is up for 00:01:16
   Authentication Sequence: [ 0 ]
   Router-ID: 3.3.3.3            Address: 172.16.1.3
   State: Full   Mode:Nbr is   Master   Priority: 1
   DR: None     BDR: None     MTU: 0
   Dead timer due in 103 sec
   Retrans timer interval: 0
   Neighbor is up for 00:01:02
   Authentication Sequence: [ 0 ]
   Router-ID: 4.4.4.4            Address: 172.16.1.4
   State: Full   Mode:Nbr is   Master   Priority: 1
   DR: None     BDR: None     MTU: 0
   Dead timer due in 113 sec
   Retrans timer interval: 0
   Neighbor is up for 00:00:57
   Authentication Sequence: [ 0 ]
```

可以观察到，DR/BDR 都为 None，验证了在点到多点的网络类型中不选举 DR/BDR，同样在点到点网络中也是，这里不再赘述。

4. 根据现网需求影响 DR/BDR 选举

现在根据需求，网络管理员要使得性能较好、处理能力较强的 R1 成为 DR，性能次之的 R2 成为 BDR，而性能最差的 R4 不能参加 DR 和 BDR 的选举，由此来完成网络的优化。

首先将 OSPF 网络类型还原为默认的广播网络类型。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]ospf network-type broadcast

[R2]interface GigabitEthernet 0/0/0
[R2-GigabitEthernet0/0/0]ospf network-type broadcast

[R3]interface GigabitEthernet 0/0/0
[R3-GigabitEthernet0/0/0]ospf network-type broadcast

[R4]interface GigabitEthernet 0/0/0
[R4-GigabitEthernet0/0/0]ospf network-type broadcast
```

配置完成后，修改 R1 上 GE 0/0/0 接口的 DR 优先级为 100、R2 为 50、R4 为 0、R3 保持默认不变。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]ospf dr-priority 100

[R2]interface GigabitEthernet 0/0/0
```

```
[R2-GigabitEthernet0/0/0]ospf dr-priority 50

[R4]interface GigabitEthernet 0/0/0
[R4-GigabitEthernet0/0/0]ospf dr-priority 0
```

配置完成后，查看各路由器的 DR/BDR 选举情况。

```
[R1]display ospf peer
          OSPF Process 1 with Router-ID 1.1.1.1
                   Neighbors
Area 0.0.0.0 interface 172.16.1.1(GigabitEthernet0/0/0)'s neighbors
Router-ID: 2.2.2.2           Address: 172.16.1.2
State: 2-Way   Mode:Nbr is   Master   Priority: 1
DR: 172.16.1.4   BDR: 172.16.1.3   MTU: 0
Dead timer due in 35   s
Retrans timer interval: 0
Neighbor is up for 00:00:00
Authentication Sequence: [ 0 ]
Router-ID: 3.3.3.3           Address: 172.16.1.3
State: Full   Mode:Nbr is   Master   Priority: 1
DR: 172.16.1.4   BDR: 172.16.1.3   MTU: 0
Dead timer due in 34   s
Retrans timer interval: 5
Neighbor is up for 00:00:48
Authentication Sequence: [ 0 ]
Router-ID: 4.4.4.4           Address: 172.16.1.4
State: Full   Mode:Nbr is   Master   Priority: 1
DR: 172.16.1.4   BDR: 172.16.1.3   MTU: 0
Dead timer due in 34   s
Retrans timer interval: 0
Neighbor is up for 00:00:46
Authentication Sequence: [ 0 ]
```

发现此时的 DR 与 BDR 都没有改变，即验证了 OSPF 的 DR/BDR 选举是非抢占的。必须要在 4 台路由器上同时重启 OSPF 进程，或者重启路由器才能使得其重新正确选举。

同时重启 4 台路由器的 OSPF 进程，或直接同时重启设备。

重置后再次查看各路由器的 DR/BDR 选举状态。

```
<R1>display ospf peer
          OSPF Process 1 with Router-ID 1.1.1.1
                   Neighbors
Area 0.0.0.0 interface 172.16.1.1(GigabitEthernet0/0/0)'s neighbors
Router-ID: 2.2.2.2           Address: 172.16.1.2
State: Full   Mode:Nbr is   Master   Priority: 50
DR: 172.16.1.1   BDR: 172.16.1.2   MTU: 0
Dead timer due in 28   s
Retrans timer interval: 5
Neighbor is up for 00:00:19
Authentication Sequence: [ 0 ]
Router-ID: 3.3.3.3           Address: 172.16.1.3
State: Full   Mode:Nbr is   Master   Priority: 1
DR: 172.16.1.1   BDR: 172.16.1.2   MTU: 0
Dead timer due in 32   s
Retrans timer interval: 5
Neighbor is up for 00:00:04
Authentication Sequence: [ 0 ]
Router-ID: 4.4.4.4           Address: 172.16.1.4
```

```
State: Full    Mode:Nbr is    Master    Priority: 0
DR: 172.16.1.1    BDR: 172.16.1.2    MTU: 0
Dead timer due in 29    s
Retrans timer interval: 5
Neighbor is up for 00:00:28
Authentication Sequence: [ 0 ]
```

此时发现在该广播网络中，R1 为 DR，R2 为 BDR，实现了网络的需求。

思考

在本实验步骤 2 中，基础的 OSPF 网络配置完毕后，为什么要同时重启 4 台路由器上的 OSPF 进程？

8.7　OSPF 开销值、协议优先级及计时器的修改

原理概述

由于路由器上可能同时运行多种动态路由协议，就存在各个路由协议之间路由信息共享和选择的问题。系统为每一种路由协议设置了不同的默认优先级，当在不同协议中发现同一条路由时，协议优先级高的将被优选。

如果没有直接配置 OSPF 接口的开销值，OSPF 会根据该接口的带宽自动计算其开销值。计算公式为：接口开销=带宽参考值/接口带宽，取计算结果的整数部分作为接口开销值（当结果小于 1 时取 1）。通过改变带宽参考值可以间接改变接口的开销值。

OSPF 常见的计时器包括 Hello timer 和 Dead timer，分别决定了 OSPF 发送 Hello 报文的间隔和保持邻居关系的计时器。默认情况下，P2P、Broadcast 类型接口发送 Hello 报文的时间间隔为 10s，邻居失效时间为 40s；P2MP、NBMA 类型接口发送 Hello 报文的时间间隔为 30s；邻居失效时间为 120s。

实验目的

- 掌握配置 OSPF 协议优先级的方法
- 掌握配置 OSPF 开销的方法
- 掌握配置 OSPF Hello timer 的方法
- 掌握配置 OSPF Dead timer 的方法

实验内容

本实验模拟企业两个分支机构之间通过两条路径实现互联互通。R1 为分支机构 A 的网关设备，R4 为分支机构 B 的网关设备。公司原网络为分支 A 与分支 B 通过 R2 进行通信，设备之间运行的是 OSPF 协议，都属于区域 0。后因带宽需要增大，两机构之间决定新增一条带宽更大路径，通过 R3 相连，运行 RIP 协议，并设置为主用路径，以前的链路为备用路径。当后期 R3 设备升级之后，可支持 OSPF 时需要将网络割接到 OSPF 协议以便于管理。

实验拓扑

OSPF 开销值、协议优先级及计时器的修改拓扑如图 8-9 所示。

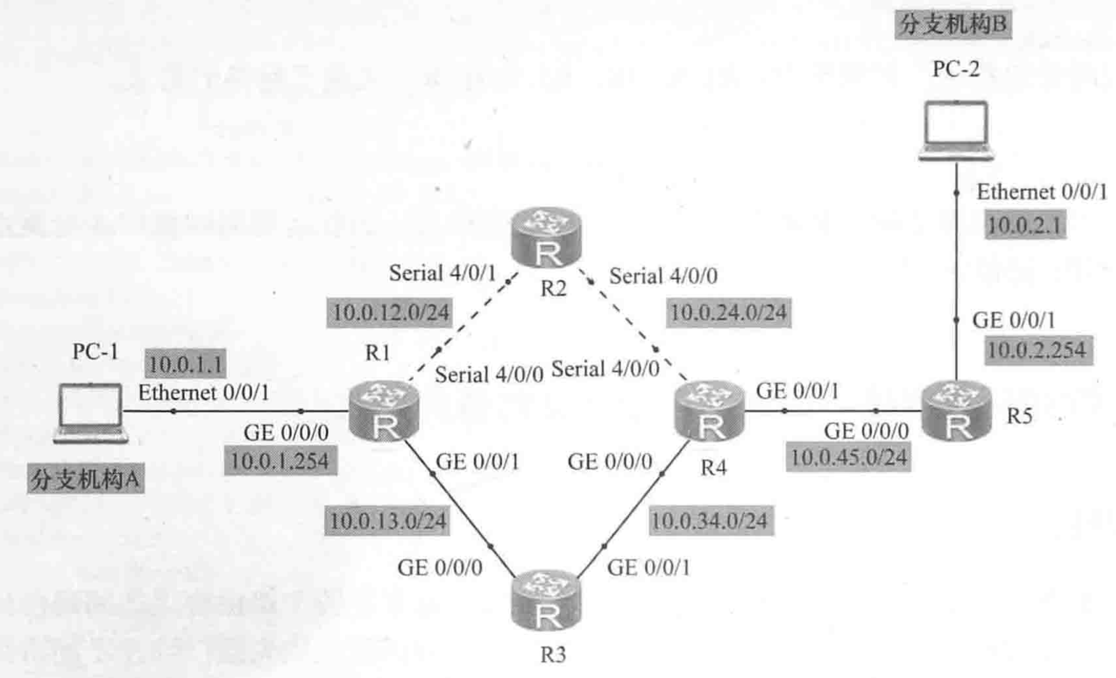


图 8-9　OSPF 开销值、协议优先级计时器的修改拓扑

实验编址

实验编址见表 8-7。

表 8-7　　实验编址

设备	接口	IP 地址	子网掩码	默认网关
PC-1	Ethernet 0/0/1	10.0.1.1	255.255.255.0	10.0.1.254
PC-2	Ethernet 0/0/1	10.0.2.1	255.255.255.0	10.0.2.254
R1（AR2220）	GE 0/0/0	10.0.1.254	255.255.255.0	N/A
	GE 0/0/1	10.0.13.1	255.255.255.0	N/A
	Serial 4/0/0	10.0.12.1	255.255.255.0	N/A
R2（AR2220）	Serial 4/0/0	10.0.24.2	255.255.255.0	N/A
	Serial 4/0/1	10.0.12.2	255.255.255.0	N/A
R3（AR2220）	GE 0/0/0	10.0.13.3	255.255.255.0	N/A
	GE 0/0/1	10.0.34.3	255.255.255.0	N/A
R4（AR2220）	GE 0/0/0	10.0.34.4	255.255.255.0	N/A
	GE 0/0/1	10.0.45.4	255.255.255.0	N/A
	Serial 4/0/0	10.0.24.4	255.255.255.0	N/A
R5（AR2220）	GE 0/0/0	10.0.45.5	255.255.255.0	N/A
	GE 0/0/1	10.0.2.254	255.255.255.0	N/A

实验步骤

1. 基本配置

根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测各直连链路的连通性。

```
<R1>ping 10.0.12.2
  PING 10.0.12.2: 56  data bytes,  press CTRL_C to break
    Reply from 10.0.12.2: bytes=56 Sequence=1 ttl=255 time=50 ms
    Reply from 10.0.12.2: bytes=56 Sequence=2 ttl=255 time=40 ms
    Reply from 10.0.12.2: bytes=56 Sequence=3 ttl=255 time=20 ms
    Reply from 10.0.12.2: bytes=56 Sequence=4 ttl=255 time=10 ms
    Reply from 10.0.12.2: bytes=56 Sequence=5 ttl=255 time=40 ms
  --- 10.0.12.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 10/32/50 ms
```

其余直连网段的连通性测试省略。

2. 配置协议优先级

部署 OSPF 网络，实现分支 A 和分支 B 之间通过 R2 实现通信。

在路由器 R1、R2、R4 上部署 OSPF 网络，通告相关网段属于区域 0。

```
[R1]ospf 1
[R1-ospf-1]area 0
[R1-ospf-1-area-0.0.0.0]network 10.0.12.0 0.0.0.255
[R1-ospf-1-area-0.0.0.0]network 10.0.1.0 0.0.0.255

[R2]ospf 1
[R2-ospf-1]area 0
[R2-ospf-1-area-0.0.0.0]network 10.0.12.0 0.0.0.255
[R2-ospf-1-area-0.0.0.0]network 10.0.24.0 0.0.0.255

[R4]ospf 1
[R4-ospf-1]area 0
[R4-ospf-1-area-0.0.0.0]network 10.0.24.0 0.0.0.255
[R4-ospf-1-area-0.0.0.0]network 10.0.45.0 0.0.0.255

[R5]ospf 1
[R5-ospf-1]area 0
[R5-ospf-1-area-0.0.0.0]network 10.0.45.0 0.0.0.255
[R5-ospf-1-area-0.0.0.0]network 10.0.2.0 0.0.0.255
```

部署完成后，测试两分支间 PC-1 与 PC-2 间的连通性。

```
PC>ping 10.0.2.1
Ping 10.0.2.1: 32 data bytes,  Press Ctrl_C to break
From 10.0.2.1: bytes=32 seq=1 ttl=124 time=47 ms
From 10.0.2.1: bytes=32 seq=2 ttl=124 time=31 ms
From 10.0.2.1: bytes=32 seq=3 ttl=124 time=47 ms
From 10.0.2.1: bytes=32 seq=4 ttl=124 time=31 ms
From 10.0.2.1: bytes=32 seq=5 ttl=124 time=47 ms
--- 10.0.2.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 31/40/47 ms
```

通信正常，即目前通过 R2 的线路正常。

现网络管理员开始实施网络升级方案，部署使用经过 R3 的线路，运行 RIP 协议。

```
[R1]rip 1
[R1-rip-1]version 2
```

```
[R1-rip-1]undo summary
[R1-rip-1]network 10.0.0.0

[R3]rip 1
[R3-rip-1]version 2
[R3-rip-1]undo summary
[R3-rip-1]network 10.0.0.0

[R4]rip 1
[R4-rip-1]version 2
[R4-rip-1]undo summary
[R4-rip-1]network 10.0.0.0

[R5]rip 1
[R5-rip-1]version 2
[R5-rip-1]undo summary
[R5-rip-1]network 10.0.0.0
```

配置完成后，在分支 A 的网关设备 R1 上查看路由表中关于分支 B 网段的 10.0.2.0 的条目。

```
[R1]display ip routing-table 10.0.2.0
Route Flags: R - relay，   D - download to fib
------------------------------------------------------------------------------
Routing Table : Public
Summary Count : 1
Destination/Mask    Proto   Pre  Cost      Flags  NextHop         Interface
10.0.2.0/24         OSPF    10   98        D      10.0.12.2       Serial4/0/0
```

发现分支 B 网段的路由条目现在仍然通过 OSPF 协议获得，即两分支间的数据仍然通过 R2 转发。新接入的 R3，带宽更大的路径没有参与数据转发，升级不成功。可以使用 **tracert** 命令在设备和 PC 上进行验证，此处省略。

导致不成功的原因是该路由条目可以同时从 OSPF 协议和 RIP 协议获得，当同一路由条目可以通过不同的路由协议获得时，首先比较两协议的优先级，路由器将优选优先级高的路由协议。OSPF 的默认协议优先级为 10，而 RIP 为 100，优先级值越低表示优先级越高，故而选择了从 OSPF 协议获得的路由条目。

但是根据实际需求，经过 R2 使用的 OSPF 线路是广域网线路，带宽很低，而经过 R3 使用的 RIP 线路是以太网线路，带宽高，所以现在一定要选择 RIP 条目进行转发。通过修改 OSPF 协议优先级即可。

在 R1、R4、R5 的进程下使用 **preference** 命令修改 OSPF 协议优先级的值为 110，大于 RIP 的 100。

```
[R1]ospf 1
[R1-ospf-1]preference 110

[R4]ospf 1
[R4-ospf-1]preference 110

[R5]ospf 1
[R5-ospf-1]preference 110
```

配置完成后，在分支 A 的网关设备 R1 上查看路由表中关于分支 B 网段的 10.0.2.0 的条目。

```
<R1>display ip routing-table 10.0.2.0
Route Flags: R - relay,   D - download to fib
------------------------------------------------------------------------------
Routing Table : Public
Summary Count : 1
Destination/Mask    Proto   Pre    Cost      Flags  NextHop        Interface
10.0.2.0/24         RIP     100    3         D      10.0.13.3      GigabitEthernet0/0/1
```

可以观察到，现在已经使用经过 R3 的线路。

在分支 B 的网关设备 R4 上查看路由表中关于分支 A 网段的 10.0.1.0 的条目。

```
<R4>display ip routing-table 10.0.1.1
Route Flags: R - relay,   D - download to fib
------------------------------------------------------------------------------
Routing Table : Public
Summary Count : 1
Destination/Mask    Proto   Pre    Cost      Flags  NextHop        Interface
10.0.1.0/24         RIP     100    2         D      10.0.34.3      GigabitEthernet0/0/0
```

R4 也采用经过 R3 的线路，往返路径一致。可以进一步使用 **tracert** 命令测试，这里省略。

3. 配置 OSPF 开销值

由于网络中运行不同路由协议将会导致管理不便，现需要更改 R3 的配置，使其运行 OSPF 协议。

在网络调整过程中最重要的就是尽量确保能够使其对用户通信所造成的影响程度降至最小，并且一般选择在用户网络使用率较少的深夜进行。经过对网络的分析后发现，在 R3 上直接部署 OSPF 协议属于区域 0 中，即和 R2 一样都运行 OSPF 协议，那么在相同 OSPF 协议下，路由的选择首先比较链路的开销值，而经过 R2 的线路为广域网链路，开销值明显高于经过 R3 的以太网链路，即仍然维持通过 R3 来转发公司两支间的流量，风险较小。故网络管理员将直接在经过 R3 的线路上部署 OSPF 协议。

在 R1、R3、R4 上配置 OSPF 协议，通告相关网段。

```
[R3]ospf 1
[R3-ospf-1]area 0
[R3-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 10.0.34.0 0.0.0.255

[R1]ospf 1
[R1-ospf-1]area 0
[R1-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255

[R4]ospf 1
[R4-ospf-1]area 0
[R4-ospf-1-area-0.0.0.0]network 10.0.34.0 0.0.0.255
```

配置完成后，在分支 A 的网关设备 R1 上查看路由表中关于分支 B 网段的 10.0.2.0 的条目。

```
<R1>display ip routing-table  10.0.2.0
Route Flags: R - relay,   D - download to fib
------------------------------------------------------------------------------
Routing Table : Public
Summary Count : 1
Destination/Mask    Proto   Pre    Cost      Flags  NextHop        Interface
10.0.2.0/24         OSPF    110    4         D      10.0.13.3      GigabitEthernet0/0/1
```

可以观察到，网络配置调整完成后，仍然维持使用 R3 的线路转发。注意，最后还要删除 RIP 协议的相关配置，以免造成不必要的隐患，删除步骤此处省略。

现要求分支机构 A 能够每月定期检查备用路径是否正常可用，那么就要求流量能够通过 R2 转发，但是由于目前经过 R2 的线路的开销值远大于经过 R3 的线路而导致无法测试，可以通过手动修改 OSPF 开销值的方法来实现路径选择。

在 R1 的 GE 0/0/1 接口上使用 **ospf cost** 命令配置运行 OSPF 协议所需的开销值。

```
[R1]interface GigabitEthernet 0/0/1
[R1-GigabitEthernet0/0/1]ospf cost 1000
```

配置完成后，在分支 A 的网关设备 R1 上查看路由表中关于分支 B 网段的 10.0.2.0 的条目。

```
<R1>display ip routing-table 10.0.2.0
Route Flags: R - relay,    D - download to fib
------------------------------------------------------------------------------
Routing Table : Public
Summary Count : 1
Destination/Mask       Proto     Pre     Cost        Flags   NextHop          Interface
10.0.2.0/24            OSPF      110     98          D       10.0.12.2        Serial4/0/0
```

可以观察到，现在发送至分支 B 的流量已经通过 R2 来转发，经过 R2 的路径的路由开销为值 98，远小于 R3 上配置的路由开销 1000。

注意，OSPF 链路开销值是基于接口修改的，一定要在路由更新的入接口修改才生效。

4. 配置 OSPF 计时器

网络管理员在日常网络巡检中发现，经过 R3 的线路是以太网，在 OSPF 协议中的网络类型为广播网络类型，即默认 Hello 计时器和 Dead 计时器是 10s 和 40s。这样 OSPF 数据的 Hello 报文发送过于频繁。

现修改 R1 上 Hello 计时器和 Dead 计时器为 20s 和 80s。

```
[R1]interface GigabitEthernet 0/0/1
[R1-GigabitEthernet0/0/1]ospf timer hello 20
[R1-GigabitEthernet0/0/1]ospf timer dead 80
```

稍等片刻，会发现 R1 与 R3 的邻居关系中断，这是因为 Hello 计时器和 Dead 计时器在 OSPF 广播网络中建立邻居关系时要进行校验，校验一致才能够建立邻居。

同样修改 R3 的两个计时器，和 R1 保持一致。

```
[R3]interface GigabitEthernet 0/0/0
[R3-GigabitEthernet0/0/1]ospf timer hello 20
[R3-GigabitEthernet0/0/1]ospf timer dead 80
```

配置完成后，查看 R1 的邻居状态。

```
<R1>display ospf peer
          OSPF Process 1 with Router-ID 10.0.1.254
                  Neighbors
Area 0.0.0.0 interface 10.0.13.1(GigabitEthernet0/0/1)'s neighbors
Router-ID: 10.0.13.3          Address: 10.0.13.3
State: Full   Mode:Nbr is   Master   Priority: 1
DR: 10.0.13.3   BDR: None     MTU: 0
Dead timer due in 79    s
Retrans timer interval: 0
```

```
Neighbor is up for 00:00:18
Authentication Sequence: [ 0 ]
```

可以观察到，邻居恢复正常。

思考

OSPF 的 Dead 计时器时长默认为什么要保持是 Hello 计时器的 4 倍？一定要保持 4 倍关系吗？

8.8 连接 RIP 与 OSPF 网络

原理概述

不同的网络会根据自身的实际情况来选用路由协议。比如有些网络规模很小，为了管理简单，部署了 RIP；而有些网络很复杂，可以部署 OSPF。不同路由协议之间不能直接共享各自的路由信息，需要依靠配置路由的引入来实现。

获得路由信息一般有 3 种途径：直连网段、静态配置和路由协议。可以将通过这 3 种途径获得的路由信息引入到路由协议中，例如，把直连网段引入到 OSPF 中，叫做“引入直连”；把静态路由引入 OSPF，叫做“引入静态路由”；把 RIP 引入 OSPF 叫做“引入 RIP”。当把这些路由信息引入到路由协议进程以后，这些路由信息就可以在路由协议进程中进行通告了，也就是说通过配置引入，一种路由协议可以自动获得所有来自另一种协议的所有路由信息。

不同的路由协议计算路由开销的依据是不同的，开销值的大小和范围都是不同的。OSPF 的开销值基于带宽，而且值的范围很大，RIP 的开销基于跳数，范围很小，所以当配置 OSPF 和 RIP 相互引入时一定要小心（在华为 VRP 平台上，当引入 OSPF 路由至 RIP 时，如不指定 Cost 值，开销值将默认设为 1。尽管如此，网络管理员还是应该手工配置开销值以反映网络的真实情况）。

实验目的

- 理解路由引入的应用场景
- 掌握 RIP 中引入其他协议的配置
- 掌握 OSPF 中引入其他协议的配置
- 掌握路由引入时修改开销值的方法

实验内容

本实验模拟真实网络场景。路由器 R1 分别连接两家公司网络，R1 左侧公司 A 内部网络运行 RIP 协议，公司 B 内部网络运行 OSPF 协议。由于业务发展需要，两家公司需要能够互相通信。但由于两家公司使用不同的路由协议，现需要在路由器 R1 上配置双向路由引入。

实验拓扑

连接 RIP 与 OSPF 网络的拓扑如图 8-10 所示。

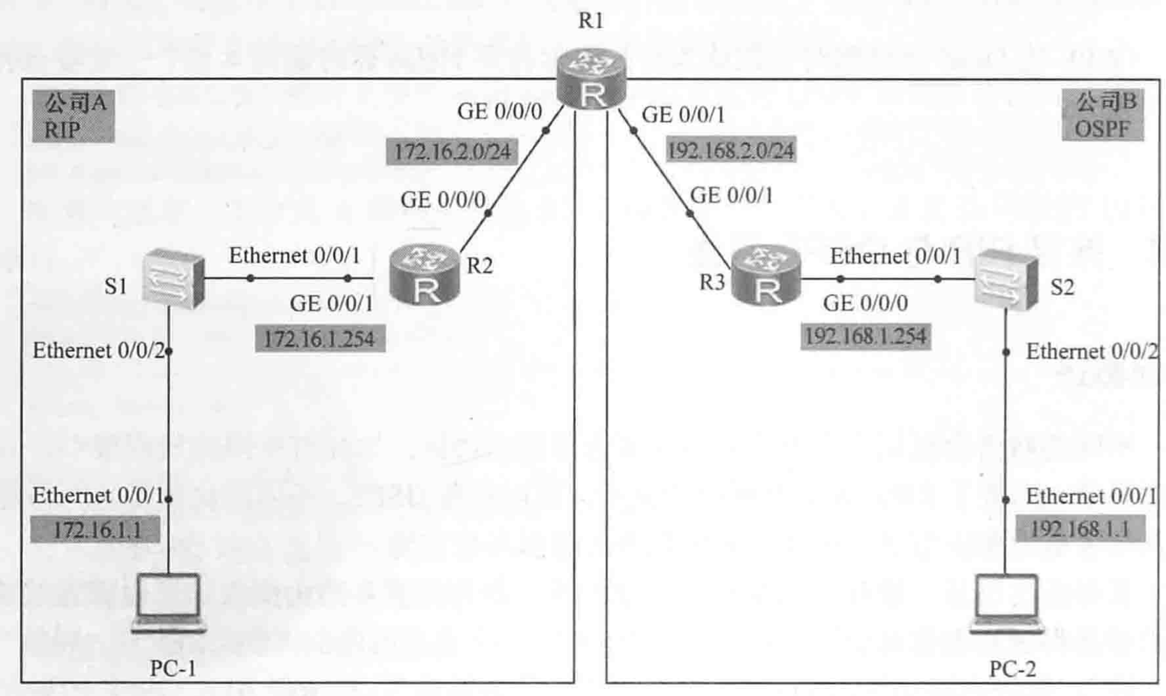


图 8-10 连接 RIP 与 OSPF 网络拓扑

实验编址

实验编址见表 8-8。

表 8-8 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR1220）	GE 0/0/0	172.16.2.1	255.255.255.0	N/A
	GE 0/0/1	192.168.2.1	255.255.255.0	N/A
R2（AR1220）	GE 0/0/0	172.16.2.2	255.255.255.0	N/A
	GE 0/0/1	172.16.1.254	255.255.255.0	N/A
R3（AR1220）	GE 0/0/0	192.168.1.254	255.255.255.0	N/A
	GE 0/0/1	192.168.2.3	255.255.255.0	N/A
PC-1	Ethernet 0/0/1	172.16.1.1	255.255.255.0	172.16.1.254
PC-2	Ethernet 0/0/1	192.168.1.1	255.255.255.0	192.168.1.254

实验步骤

1. 基本配置

根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测各直连链路的连通性。

```
[R2]ping 172.16.1.1
  PING 172.16.1.1: 56   data bytes, press CTRL_C to break
    Reply from 172.16.1.1: bytes=56 Sequence=1 ttl=128 time=50 ms
    Reply from 172.16.1.1: bytes=56 Sequence=2 ttl=128 time=60 ms
    Reply from 172.16.1.1: bytes=56 Sequence=3 ttl=128 time=80 ms
```

```
    Reply from 172.16.1.1: bytes=56 Sequence=4 ttl=128 time=30 ms
    Reply from 172.16.1.1: bytes=56 Sequence=5 ttl=128 time=60 ms
  --- 172.16.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 30/56/80 ms
```

其余直连网段的连通性测试省略。

2. 搭建 RIP 和 OSPF 网络

根据图 8-10 配置路由协议，公司 A 内部运行 RIP 协议。在 R1 和 R2 上配置 RIP，进程号为 1，启用 RIP v2 版本、关闭自动汇总，通告各自接口所在网段，R1 在 RIP 中仅通告 GE 0/0/0 接口所在网段。

```
[R1]rip 1
[R1-rip-1]version 2
[R1-rip-1]undo summary
[R1-rip-1]network 172.16.0.0

[R2]rip 1
[R2-rip-1]version 2
[R2-rip-1]undo summary
[R2-rip-1]network 172.16.0.0
```

公司 B 内部运行 OSPF 协议。在 R1 和 R3 上配置 OSPF，使用进程号 1，所有网段都属于区域 0，R1 在 OSPF 中仅通告 GE 0/0/1 接口所在网段。

```
[R1]ospf 1
[R1-ospf-1]area 0
[R1-ospf-1-area-0.0.0.0]network 192.168.2.0 0.0.0.255

[R3]ospf 1
[R3-ospf-1]area 0
[R3-ospf-1]network 192.168.2.0 0.0.0.255
[R3-ospf-1]network 192.168.1.0 0.0.0.255
```

配置完成后查看 R1 的路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8          Routes : 8
Destination/Mask    Proto   Pre   Cost    Flags   NextHop       Interface
     172.16.1.0/24  RIP     100   1       D       172.16.2.2    GigabitEthernet0/0/0
     172.16.2.0/24  Direct  0     0       D       172.16.2.1    GigabitEthernet0/0/0
     172.16.2.1/32  Direct  0     0       D       127.0.0.1     GigabitEthernet0/0/0
    192.168.1.0/24  OSPF    10    2       D       192.168.2.3   GigabitEthernet0/0/1
    192.168.2.0/24  Direct  0     0       D       192.168.2.1   GigabitEthernet0/0/1
    192.168.2.1/32  Direct  0     0       D       127.0.0.1     GigabitEthernet0/0/1
       127.0.0.0/8  Direct  0     0       D       127.0.0.1     InLoopBack0
      127.0.0.1/32  Direct  0     0       D       127.0.0.1     InLoopBack0
```

由于 R1 上同时运行了 RIP 协议和 OSPF 协议，可以观察到 R1 同时拥有公司 A 和公司 B 的路由信息。

3. 配置双向路由引入

为了使两个公司网络能够互相访问，需要把公司 A 的 RIP 协议的路由引入到公司 B 的 OSPF 协议中，同样把公司 B 的 OSPF 协议的路由引入到公司 A 的 RIP 协议中。

在 R1 的 OSPF 进程中使用 **import-route rip** 命令引入 RIP 路由。

```
[R1]ospf 1
[R1-ospf-1]import-route rip 1
```

配置完成后，查看 R3 的路由表。

```
[R3]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8        Routes : 8
Destination/Mask    Proto   Pre   Cost    Flags   NextHop         Interface
     172.16.1.0/24  O_ASE   150   1       D       192.168.2.1     GigabitEthernet0/0/1
     172.16.2.0/24  O_ASE   150   1       D       192.168.2.1     GigabitEthernet0/0/1
    192.168.1.0/24  Direct  0     0       D       192.168.1.254   GigabitEthernet0/0/0
  192.168.1.254/32  Direct  0     0       D       127.0.0.1       GigabitEthernet0/0/0
    192.168.2.0/24  Direct  0     0       D       192.168.2.3     GigabitEthernet0/0/1
    192.168.2.3/32  Direct  0     0       D       127.0.0.1       GigabitEthernet0/0/1
       127.0.0.0/8  Direct  0     0       D       127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0     0       D       127.0.0.1       InLoopBack0
```

可以观察到 R3 上现在拥有来自公司 A 的路由信息。

在 R1 的 RIP 进程中使用 **import-route ospf** 命令引入 OSPF 路由。

```
[R1]rip 1
[R1-rip-1]import-route ospf 1
```

配置完成后，查看 R2 的路由表。

```
[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8        Routes : 8
Destination/Mask    Proto   Pre   Cost    Flags   NextHop         Interface
     172.16.1.0/24  Direct  0     0       D       172.16.1.254    GigabitEthernet0/0/1
   172.16.1.254/32  Direct  0     0       D       127.0.0.1       GigabitEthernet0/0/1
     172.16.2.0/24  Direct  0     0       D       172.16.2.2      GigabitEthernet0/0/0
     172.16.2.2/32  Direct  0     0       D       127.0.0.1       GigabitEthernet0/0/0
    192.168.1.0/24  RIP     100   1       D       172.16.2.1      GigabitEthernet0/0/0
    192.168.2.0/24  RIP     100   1       D       172.16.2.1      GigabitEthernet0/0/0
       127.0.0.0/8  Direct  0     0       D       127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0     0       D       127.0.0.1       InLoopBack0
```

可以观察到 R2 上现在拥有来自公司 B 的路由信息，且路由的开销值默认都为 1。

当配置路由引入后双方可以互相获得对方的路由信息，但是在各自的路由表中，开销都为默认值 1。

4. 手工配置引入时的开销值

为了能够反映真实的网络拓扑情况，更好地进行路由控制。网络管理员在将 OSPF 引入 RIP 时手工配置路由开销值，例如在 R1 的 RIP 进程中使用 **import-route ospf 1 cost 3** 命令修改开销值为 3。

```
[R1]rip 1
[R1-rip-1]import-route ospf 1 cost 3
```

配置完成后，在 R2 上查看 Cost 值的变化情况。

```
[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
        Destinations : 10        Routes : 10
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
      172.16.1.0/24  Direct  0    0          D   172.16.1.254    GigabitEthernet0/0/1
    172.16.1.254/32  Direct  0    0          D   127.0.0.1       GigabitEthernet0/0/1
      172.16.2.0/24  Direct  0    0          D   172.16.2.2      GigabitEthernet0/0/0
      172.16.2.2/32  Direct  0    0          D   127.0.0.1       GigabitEthernet0/0/0
     192.168.1.0/24  RIP     100  4          D   172.16.2.1      GigabitEthernet0/0/0
     192.168.2.0/24  RIP     100  4          D   172.16.2.1      GigabitEthernet0/0/0
        127.0.0.0/8  Direct  0    0          D   127.0.0.1       InLoopBack0
       127.0.0.1/32  Direct  0    0          D   127.0.0.1       InLoopBack0
```

可以观察到，在 R2 的路由表中两条路由的 Cost 值已经变为 4，这是因为还加上了 R2 接口上的 Cost 值 1。

思考

关于在路由引入时手工修改路由的 Cost 值，这么做还有其他的用处吗？

8.9 使用 RIP、OSPF 发布默认路由

原理概述

默认路由是指目的地址和掩码都是 0 的路由条目。当路由器无精确匹配的路由时，就可以通过默认路由进行报文转发。

合理使用默认路由，可以在很大程度上减小本地路由表的大小，节约设备资源。默认路由可以在路由器上手工配置，也可以由路由协议自动发布。

RIP 和 OSPF 这两种路由协议都可以通过配置使路由器对协议邻居发布默认路由，并且可以设置该路由的度量值。

实验目的

- 理解默认路由的应用场景
- 掌握 RIP 发布默认路由的配置
- 掌握 OSPF 发布默认路由的配置

实验内容

本实验模拟真实网络场景。路由器 R1 分别连接两家公司网络，R1 左侧公司 A 内部网络运行 RIP 协议，公司 B 内部网络运行 OSPF 协议。由于业务发展需要，两家公司人员需要能够互相通信，但是为了保护自身网络的私密性，双方都不愿意对方知道自己网络的明细路由。这种情况下需要配置路由协议以自动发布默认路由的方式来完成此需求。

实验拓扑

使用 RIP、OSPF 发布默认路由的拓扑如图 8-11 所示。

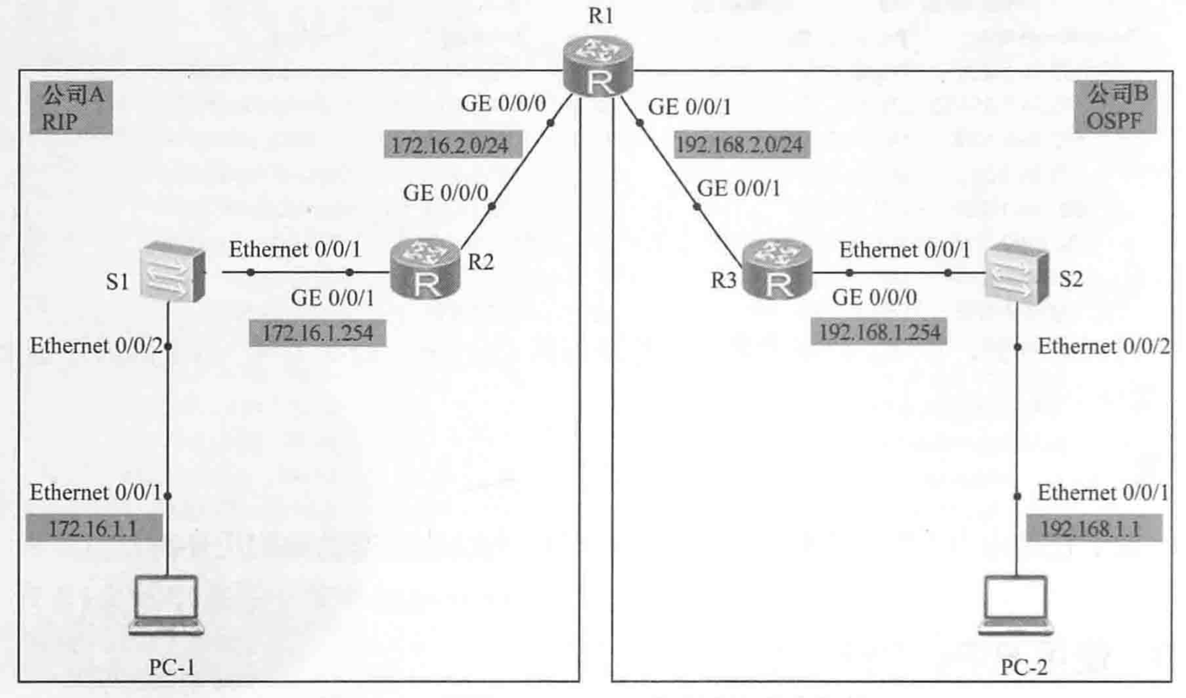


图 8-11　使用 RIP、OSPF 发布默认路由拓扑

实验编址

实验编址见表 8-9。

表 8-9　　实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR1220）	GE 0/0/0	172.16.2.1	255.255.255.0	N/A
	GE 0/0/1	192.168.2.1	255.255.255.0	N/A
R2（AR1220）	GE 0/0/0	172.16.2.2	255.255.255.0	N/A
	GE 0/0/1	172.16.1.254	255.255.255.0	N/A
R3（AR1220）	GE 0/0/0	192.168.1.254	255.255.255.0	N/A
	GE 0/0/1	192.168.2.3	255.255.255.0	N/A
PC-1	Ethernet 0/0/1	172.16.1.1	255.255.255.0	172.16.1.254
PC-2	Ethernet 0/0/1	192.168.1.1	255.255.255.0	192.168.1.254

实验步骤

1．基本配置

根据实验编址表进行相应的基本 IP 地址配置，并使用 **ping** 命令检测各直连链路的连通性。

```
[R2]ping 172.16.1.1
  PING 172.16.1.1: 56  data bytes, press CTRL_C to break
    Reply from 172.16.1.1: bytes=56 Sequence=1 ttl=128 time=50 ms
    Reply from 172.16.1.1: bytes=56 Sequence=2 ttl=128 time=60 ms
```

```
    Reply from 172.16.1.1: bytes=56 Sequence=3 ttl=128 time=80 ms
    Reply from 172.16.1.1: bytes=56 Sequence=4 ttl=128 time=30 ms
    Reply from 172.16.1.1: bytes=56 Sequence=5 ttl=128 time=60 ms
  --- 172.16.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 30/56/80 ms
```

其余直连网段的连通性测试省略。

2. 配置 RIP 和 OSPF 路由协议

根据实验拓扑图配置路由协议，公司 A 内部运行 RIP 协议。在 R1 和 R2 上配置 RIP，进程号为 1，启用 RIP v2 版本、关闭自动汇总，通告各自接口所在网段，R1 在 RIP 中仅通告 GE 0/0/0 接口所在网段。

```
[R1]rip 1
[R1-rip-1]version 2
[R1-rip-1]undo summary
[R1-rip-1]network 172.16.0.0

[R2]rip 1
[R2-rip-1]version 2
[R2-rip-1]undo summary
[R2-rip-1]network 172.16.0.0
```

公司 B 内部运行 OSPF 协议。在 R1 和 R3 上配置 OSPF，使用进程号 1，所有网段都属于区域 0，R1 在 OSPF 中仅通告 GE 0/0/1 接口所在网段。

```
[R1]ospf 1
[R1-ospf-1]area 0
[R1-ospf-1-area-0.0.0.0]network 192.168.2.0 0.0.0.255

[R3]ospf 1
[R3-ospf-1]area 0
[R3-ospf-1]network 192.168.2.0 0.0.0.255
[R3-ospf-1]network 192.168.1.0 0.0.0.255
```

配置完成后查看 R1 的路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8        Routes : 8
Destination/Mask    Proto   Pre    Cost     Flags   NextHop        Interface
      172.16.1.0/24  RIP     100    1        D       172.16.2.2     GigabitEthernet0/0/0
      172.16.2.0/24  Direct  0      0        D       172.16.2.1     GigabitEthernet0/0/0
      172.16.2.1/32  Direct  0      0        D       127.0.0.1      GigabitEthernet0/0/0
     192.168.1.0/24  OSPF    10     2        D       192.168.2.3    GigabitEthernet0/0/1
     192.168.2.0/24  Direct  0      0        D       192.168.2.1    GigabitEthernet0/0/1
     192.168.2.1/32  Direct  0      0        D       127.0.0.1      GigabitEthernet0/0/1
     127.0.0.0/8     Direct  0      0        D       127.0.0.1      InLoopBack0
     127.0.0.1/32    Direct  0      0        D       127.0.0.1      InLoopBack0
```

由于 R1 上同时运行了 RIP 协议和 OSPF 协议，可以观察到 R1 同时拥有公司 A 和公司 B 的路由信息。

查看 R2 和 R3 的路由表。

```
[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
```

```
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 6        Routes : 6
Destination/Mask    Proto   Pre  Cost   Flags   NextHop         Interface
     172.16.1.0/24  Direct  0    0      D       172.16.1.254    GigabitEthernet0/0/1
   172.16.1.254/32  Direct  0    0      D       127.0.0.1       GigabitEthernet0/0/1
     172.16.2.0/24  Direct  0    0      D       172.16.2.2      GigabitEthernet0/0/0
     172.16.2.2/32  Direct  0    0      D       127.0.0.1       GigabitEthernet0/0/0
       127.0.0.0/8  Direct  0    0      D       127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0      D       127.0.0.1       InLoopBack0

[R3]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 6        Routes : 6
Destination/Mask    Proto   Pre  Cost   Flags   NextHop         Interface
    192.168.1.0/24  Direct  0    0      D       192.168.1.254   GigabitEthernet0/0/0
  192.168.1.254/32  Direct  0    0      D       127.0.0.1       GigabitEthernet0/0/0
    192.168.2.0/24  Direct  0    0      D       192.168.2.3     GigabitEthernet0/0/1
    192.168.2.3/32  Direct  0    0      D       127.0.0.1       GigabitEthernet0/0/1
       127.0.0.0/8  Direct  0    0      D       127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0      D       127.0.0.1       InLoopBack0
```

可以观察到，此时在 R2 和 R3 上都只拥有本公司的路由信息。测试 PC-1 与 PC-2 间的连通性。

```
PC>ping 192.168.1.1
Ping 192.168.1.1: 32 data bytes, Press Ctrl_C to break
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
......
```

可以观察到，公司 A 和公司 B 之间的 PC 也无法进行通信。

3. 配置 RIP 发布默认路由

公司 A 需要能访问公司 B 的网络，而公司 B 为了保护自身网络私密性，不希望公司 A 获知自身内部网络的明细路由，这时可以在 R1 的 RIP 协议进程中发布默认路由，使公司 A 能在没有公司 B 的明细路由的情况下访问公司 B 的网络。

在 R1 的 RIP 进程中，使用**default-route originate**命令发布默认路由。

```
[R1]rip 1
[R1-rip-1]default-route originate
```

配置完成后，在 R2 上查看路由表。

```
[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7        Routes : 7
Destination/Mask    Proto   Pre  Cost   Flags   NextHop         Interface
         0.0.0.0/0  RIP     100  1      D       172.16.2.1      GigabitEthernet0/0/0
     172.16.1.0/24  Direct  0    0      D       172.16.1.254    GigabitEthernet0/0/1
   172.16.1.254/32  Direct  0    0      D       127.0.0.1       GigabitEthernet0/0/1
     172.16.2.0/24  Direct  0    0      D       172.16.2.2      GigabitEthernet0/0/0
     172.16.2.2/32  Direct  0    0      D       127.0.0.1       GigabitEthernet0/0/0
```

```
        127.0.0.0/8    Direct   0     0      D     127.0.0.1       InLoopBack0
       127.0.0.1/32    Direct   0     0      D     127.0.0.1       InLoopBack0
```

可以观察到 R2 上有一条从 RIP 协议获取来的默认路由，通过这条默认路由，公司 A 可以访问公司 B 的网络。

4. 配置 OSPF 发布默认路由

为了能够实现通信，公司 B 也需要访问公司 A 的网络，而公司 A 同样为了保护自身网络私密性，不希望公司 B 获知自身内部网络的明细路由。这时可以在 R1 的 OSPF 协议进程中发布默认路由，使公司 B 能在没有公司 A 的明细路由的情况下能够访问公司 A 的网络。

在 R1 的 OSPF 进程中，使用 **default-route-advertise always** 命令发布默认路由。

```
[R1]ospf 1
[R1-ospf-1]default-route-advertise always
```

配置完成后，在 R3 上查看路由表。

```
[R3]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7         Routes : 7
Destination/Mask    Proto   Pre  Cost   Flags  NextHop         Interface
        0.0.0.0/0   O_ASE   150  1       D     192.168.2.1     GigabitEthernet0/0/1
     192.168.1.0/24  Direct  0    0       D     192.168.1.254   GigabitEthernet0/0/0
   192.168.1.254/32  Direct  0    0       D     127.0.0.1       GigabitEthernet0/0/0
     192.168.2.0/24  Direct  0    0       D     192.168.2.3     GigabitEthernet0/0/1
     192.168.2.3/32  Direct  0    0       D     127.0.0.1       GigabitEthernet0/0/1
        127.0.0.0/8  Direct  0    0       D     127.0.0.1       InLoopBack0
       127.0.0.1/32  Direct  0    0       D     127.0.0.1       InLoopBack0
```

可以观察到 R3 上有一条通过 OSPF 协议获得的默认路由，通过这条默认路由，公司 B 可以访问公司 A 的网络。

再次验证 PC-1 与 PC-2 之间的连通性。

```
PC>ping 192.168.1.1
Ping 192.168.1.1: 32 data bytes, Press Ctrl_C to break
From 192.168.1.1: bytes=32 seq=1 ttl=125 time=109 ms
From 192.168.1.1: bytes=32 seq=2 ttl=125 time=78 ms
From 192.168.1.1: bytes=32 seq=3 ttl=125 time=78 ms
From 192.168.1.1: bytes=32 seq=4 ttl=125 time=94 ms
From 192.168.1.1: bytes=32 seq=5 ttl=125 time=62 ms
--- 192.168.1.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 62/84/109 ms
```

通过观察可以看到此时 PC 之间能够正常通信。

通过配置看到在 RIP 和 OSPF 中都可以为各自路由协议发布默认路由。配置默认路由在可以保证网络的可达性的情况下，不仅可以保护网络的私密性，同时能够有效减少路由表中路由条目的数量，使得路由器不需要维护大量的路由信息，同时其配置和维护相对简单。

思考

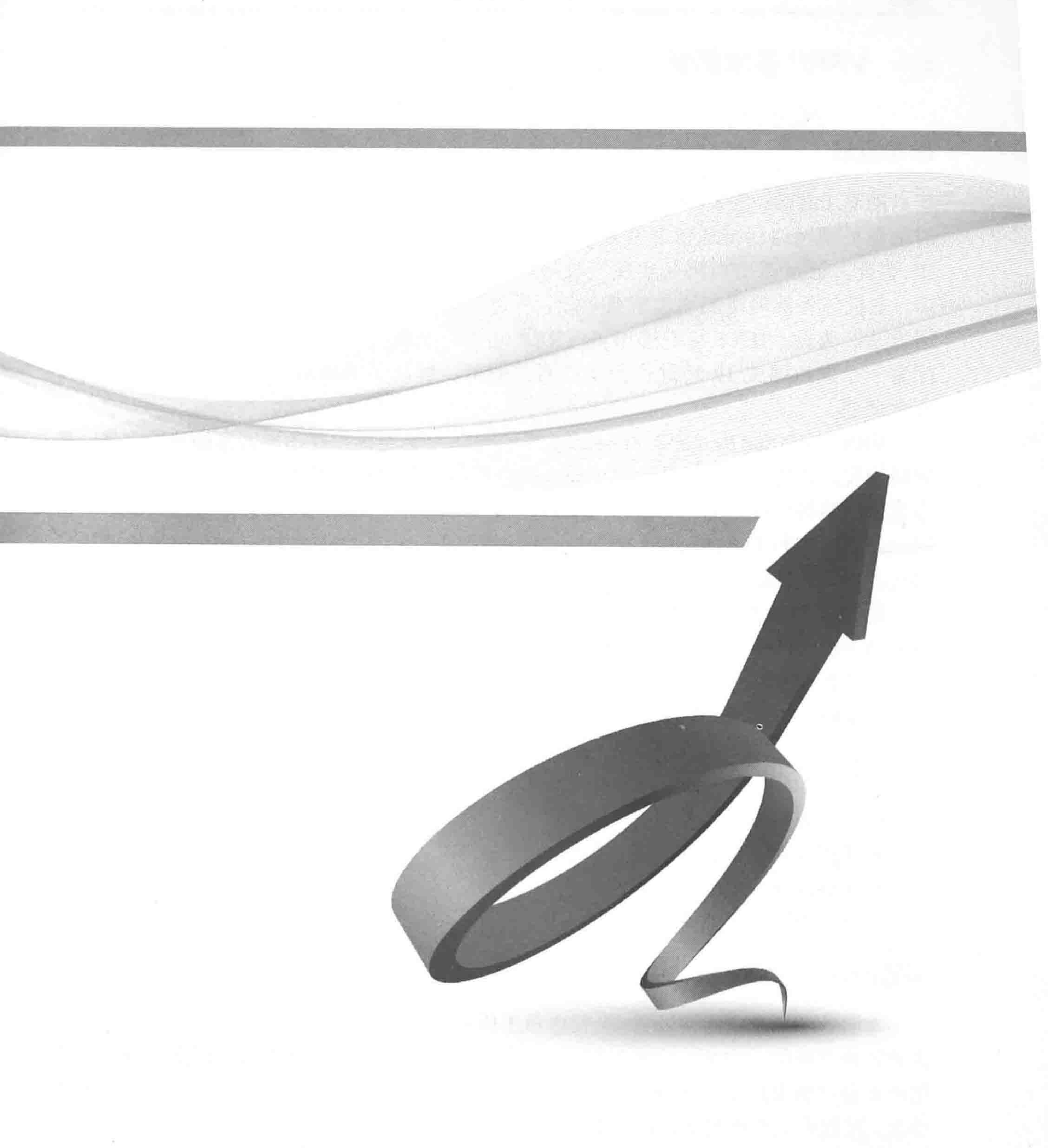
在本实验的步骤 4 中，OSPF 发布默认路由时使用到了 **default-route advertise always** 命令，如果末尾不加 always 参数，会出现什么情况？如何解决？

第9章 VRRP

9.1 VRRP基本配置

9.2 配置VRRP多备份组

9.3 配置VRRP的跟踪接口及认证



9.1 VRRP 基本配置

原理概述

随着 Internet 的发展，人们对网络可靠性的要求越来越高。对于用户来说，能够时刻与外部网络保持通信非常重要，但内部网络中的所有主机通常只能设置一个网关 IP 地址，通过该出口网关实现主机与外部网络的通信。若此时出口网关设备发生故障，主机与外部网络的通信就会中断，所以配置多个出口网关是提高网络可靠性的常用方法。为此，IETF 组织推出了 VRRP 协议，主机在多个出口网关的情况下，仅需配置一个虚拟网关 IP 地址作为出口网关即可，解决了局域网主机访问外部网络的可靠性问题。

VRRP（Virtual Router Redundancy Protocol）全称是虚拟路由器冗余协议，它是一种容错协议。该协议通过把几台路由设备联合组成一台虚拟的路由设备，该虚拟路由器在本地局域网拥有唯一的一个虚拟 ID 和虚拟 IP 地址。实际上，该虚拟路由器是由一个 Master 设备和若干 Backup 设备组成。正常情况下，业务全部由 Master 承担，所有用户端仅需设置此虚拟 IP 为网关地址。当 Master 出现故障时，Backup 接替工作，及时将业务切换到备份路由器，从而保持通信的连续性和可靠性。而用户端无需做任何配置更改，对故障无感知。

VRRP 的 Master 选举基于优先级，优先级取值范围是 0~255，默认情况下，配置优先级为 100。在接口上可以通过配置优先级的大小来手工选择 Master 设备。

实验目的

- 理解 VRRP 的应用场景
- 掌握 VRRP 虚拟路由器的配置
- 掌握修改 VRRP 优先级的方法
- 掌握查看 VRRP 主备状态的方法

实验内容

本实验模拟企业网络场景。公司内员工所用电脑，如 PC-1、PC-2，通过交换机 LSW1 连接到公司网络，LSW1 连接到公司出口网关路由器。为了提高网络的可靠性，公司使用两台路由器 R2 与 R3 作为双出口连接到外网路由器 R1。R1、R2、R3 之间运行 OSPF 协议。在双网关的情况下，如果在 PC 上配置 R2 或 R3 的真实 IP 地址作为网关，当其中一台路由器故障时，就需要手动更改 PC 的网关 IP，若网络中有大量 PC 则需要耗费大量时间和人力去更改配置，且会带来一定时间的断网影响。为了能够使故障所造成的断网影响达到最小化，增强网络的可靠性，网络管理员在 R2 与 R3 之间部署 VRRP 协议，这样当任一网关发生故障时就能自动切换而无需更改 PC 的网关 IP 地址。

实验拓扑

VRRP 基本配置的拓扑如图 9-1 所示。

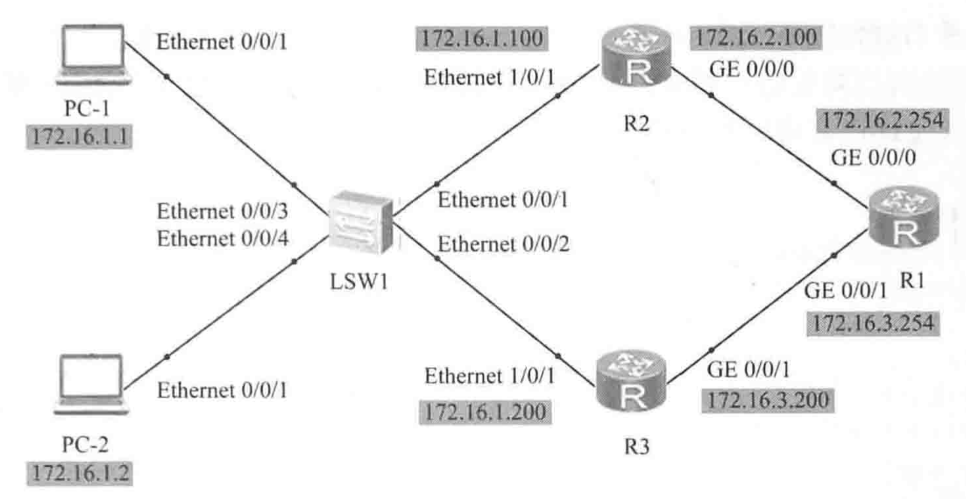


图 9-1 VRRP 基本配置拓扑

实验编址

实验编址见表 9-1。

表 9-1 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR1220）	GE 0/0/0	172.16.2.254	255.255.255.0	N/A
	GE 0/0/1	172.16.3.254	255.255.255.0	N/A
R2（AR1220）	GE 0/0/0	172.16.2.100	255.255.255.0	N/A
	Ethernet 1/0/1	172.16.1.100	255.255.255.0	N/A
R3（AR1220）	GE 0/0/1	172.16.3.200	255.255.255.0	N/A
	Ethernet 1/0/1	172.16.1.200	255.255.255.0	N/A
PC-1	Ethernet 0/0/1	172.16.1.1	255.255.255.0	172.16.1.254
PC-2	Ethernet 0/0/1	172.16.1.2	255.255.255.0	172.16.1.254

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 地址配置，并使用 **ping** 命令检测各直连链路的连通性。

```
[R1]ping 172.16.2.100
  PING 172.16.2.100: 56   data bytes, press CTRL_C to break
    Reply from 172.16.2.100: bytes=56 Sequence=1 ttl=255 time=110 ms
    Reply from 172.16.2.100: bytes=56 Sequence=2 ttl=255 time=10 ms
    Reply from 172.16.2.100: bytes=56 Sequence=3 ttl=255 time=1 ms
    Reply from 172.16.2.100: bytes=56 Sequence=4 ttl=255 time=20 ms
    Reply from 172.16.2.100: bytes=56 Sequence=5 ttl=255 time=50 ms
  --- 172.16.2.100 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 1/38/110 ms
```

其余直连网段的连通性测试省略。

2. 部署 OSPF 网络

在公司的出口网关路由器 R1、R2 和外网路由器 R3 上配置 OSPF 协议，使用进程号 1，且所有网段均通告进区域 0 中。

```
[R1]ospf 1
[R1-ospf-1]area 0
[R1-ospf-1-area-0.0.0.0]network 172.16.2.0 0.0.0.255
[R1-ospf-1-area-0.0.0.0]network 172.16.3.0 0.0.0.255

[R2]ospf 1
[R2-ospf-1]area 0
[R2-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255
[R2-ospf-1-area-0.0.0.0]network 172.16.2.0 0.0.0.255

[R3]ospf 1
[R3-ospf-1]area 0
[R3-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 172.16.3.0 0.0.0.255
```

配置完成后，在 R1 上检查 OSPF 邻居建立情况。

```
[R1]display ospf peer brief
        OSPF Process 1 with Router-ID 172.16.2.254
                Peer Statistic Information
----------------------------------------------------------------------------
 Area Id          Interface                    Neighbor id       State
 0.0.0.0          GigabitEthernet0/0/0         172.16.1.100      Full
 0.0.0.0          GigabitEthernet0/0/1         172.16.1.200      Full
```

可以观察到，此时 R1 已经与 R2、R3 成功建立起了 OSPF 邻居关系。

3. 配置 VRRP 协议

为了提高网络的可靠性，公司采用双出口的方式连接到外网。现网络管理员想针对两台出口网关路由器实现主备备份，即正常情况下，只有主网关工作，当其发生故障时能够自动切换到备份网关。现在通过配置 VRRP 协议来实现这样的需求。

在 R2 和 R3 上配置 VRRP 协议，使用 **vrrp vrid 1 virtual-ip** 命令创建 VRRP 备份组，指定即 R1 和 R2 处于同一个 VRRP 备份组内，VRRP 备份组号为 1，配置虚拟 IP 为 172.16.1.254。注意虚拟 IP 地址必须和当前接口在同一网段。

```
[R2]interface ethernet1/0/1
[R2-Ethernet1/0/1]vrrp vrid 1 virtual-ip 172.16.1.254

[R3]interface ethernet1/0/1
[R3-Ethernet1/0/1]vrrp vrid 1 virtual-ip 172.16.1.254
```

经过配置后，PC 将使用虚拟路由器 IP 地址作为默认网关。

在 VRRP 协议中，优先级决定路由器在备份组中的角色，优先级高者成为 Master。如果优先级相同，比较接口的 IP 地址大小，较大的成为 Master。优先级值默认为 100，0 被系统保留，255 保留给 IP 地址拥有者使用。

现在配置 R2 的优先级为 120，R3 的优先级保持默认 100 不变，这将使得 R2 成为 Master，R3 为 Backup。

```
[R2-Ethernet1/0/1]vrrp vrid 1 priority 120
```

配置完成后，在 R2 和 R3 上使用 **display vrrp** 命令查看 VRRP 信息。

```
[R2]display vrrp
Ethernet1/0/1 | Virtual Router 1
State : Master
Virtual IP : 172.16.1.254
Master IP : 172.16.1.100
PriorityRun : 120
PriorityConfig : 120
MasterPriority : 120
Preempt : YES     Delay Time : 0 s
……

[R3]display vrrp
Ethernet1/0/1 | Virtual Router 1
State : Backup
Virtual IP : 172.16.1.254
Master IP : 172.16.1.100
PriorityRun : 100
PriorityConfig : 100
MasterPriority : 120
Preempt : YES     Delay Time : 0 s
……
```

可以观察到现在 R2 的 VRRP 状态是 Master，R3 是 Backup。两者都处在 VRRP 备份组 1 中，且都是 E 1/0/1 接口运行在 VRRP 协议中。输出信息中的 PriorityRun 表示设备当前的运行优先级；PriorityConfig 表示为该设备配置的优先级；MasterPriority 为该备份组中 Master 的优先级；一般配置优先级就是运行优先级，但个别情况下可能运行优先级和配置优先级会不一样，在后续实验中会进行讨论。

也可以使用 **display vrrp brief** 或 **display vrrp interface** 命令来显示 VRRP 的工作状态，以 R2 为例。

```
[R2]display vrrp brief
VRID  State       Interface                Type     Virtual IP
1     Master      Eth1/0/1                 Normal   172.16.1.254
----------------------------------------------------------------------------
Total:1     Master:1     Backup:0     Non-active:0

[R2]display vrrp interface ethernet1/0/1
Ethernet1/0/1 | Virtual Router 1
State : Master
Virtual IP : 172.16.1.254
Master IP : 172.16.1.100
PriorityRun : 120
PriorityConfig : 120
MasterPriority : 120
Preempt : YES     Delay Time : 0 s
……
```

测试 PC 访问公网时的数据包转发路径。

```
PC>tracert 172.16.2.254
traceroute to 172.16.2.254, 8 hops max
(ICMP), press Ctrl+C to stop
```

```
 1  172.16.1.100    32 ms   31 ms   15 ms
 2  172.16.2.254    63 ms   62 ms   47 ms
```

可以观察此时都是通过 R2 转发。

4. 验证 VRRP 主备切换

现在手动模拟网络出现故障，将 LSW1 的 E 0/0/1 接口关闭。

```
[LSW1]interface ethernet0/0/1
[LSW1-Ethernet0/0/1]shutdown
```

经过 3s 左右，使用 **display vrrp** 查看 R3 的 VRRP 信息。

```
[R3]display vrrp
  Ethernet1/0/1 | Virtual Router 1
State : Master
    Virtual IP : 172.16.1.254
    Master IP : 172.16.1.200
PriorityRun : 100
PriorityConfig : 100
MasterPriority : 100
    Preempt : YES     Delay Time : 0 s
……
```

可以观察到 R3 切换成为了 Master，从而能够确保用户对公网的访问，几乎感知不到故障的发生。

测试 PC 访问公网时的数据包转发路径。

```
PC>tracert 172.16.2.254
traceroute to 172.16.2.254, 8 hops max
(ICMP), press Ctrl+C to stop
 1  172.16.1.200    63 ms   15 ms   32 ms
 2  172.16.2.254    62 ms   62 ms   32 ms
```

发现数据包发送路径已经切换到 R3。

如果 R2 从故障中恢复，手动开启 LSW1 的 E 0/0/1 接口。

```
[LWS1]interface ethernet0/0/1
[LSW1-Ethernet0/0/1]undo shutdown
```

查看 R2 和 R3 的 VRRP 工作状态。

```
[R2]display vrrp brief
Total:1      Master:1      Backup:0      Non-active:0
VRID   State         Interface                Type       Virtual IP
1      Master        Eth1/0/1                 Normal     172.16.1.254

[R3]display vrrp brief
Total:1      Master:1      Backup:0      Non-active:0
VRID   State         Interface                Type       Virtual IP
1      Backup        Eth1/0/1                 Normal     172.16.1.254
```

可以观察到 Master 设备又立刻重新切换回至 R2。

测试 PC 访问公网时的数据包转发路径。

```
PC>tracert 172.16.2.254
traceroute to 172.16.2.254, 8 hops max
(ICMP), press Ctrl+C to stop
1  172.16.1.100    47 ms   47 ms   46 ms
2  172.16.2.254    78 ms   78 ms   62 ms
```

可以验证也切换回 R2 转发。而这整个过程对于用户来说是透明的。

思考

如果主路由器出现故障，比如断电停机了，备份路由器是通过什么机制检测到的？

9.2 配置 VRRP 多备份组

原理概述

当 VRRP 配置为单备份组时，业务全部由 Master 设备承担，而 Backup 设备完全处于空闲状态，没有得到充分利用。VRRP 可以通过配置多备份组来实现负载分担，有效地解决了这一问题。

VRRP 允许同一台设备的同一个接口加入多个 VRRP 备份组，在不同备份组中有不同的优先级，使得各备份组中的 Master 设备不同，也就是建立多个虚拟网关路由器。各主机可以使用不同的虚拟组路由器作为网关出口，这样可以达到分担数据流而又相互备份的目的，充分利用了每一台设备的资源。

VRRP 的优先级取值范围中，255 是保留给 IP 地址拥有者使用的，当一个 VRRP 路由器的物理端口 IP 地址和虚拟路由器的虚拟 IP 地址相同，这台路由器称为虚拟 IP 地址拥有者，VRRP 优先级自动设置为 255；优先级 0 也是特殊值，当 Master 设备删除 VRRP 配置停止运行 VRRP 时，会发送优先级为 0 的 VRRP 报文通知 Backup 设备，当 Backup 收到该消息后，立刻从 Backup 状态转为 Master 状态。

实验目的

- 理解 VRRP 多备份组的应用场景
- 掌握 VRRP 多备份组的配置方法
- 理解 VRRP 的运行优先级和配置优先级
- 理解 VRRP 虚拟地址拥有者的应用

实验内容

本实验模拟企业网络场景。该公司使用两台路由器 R2 和 R3 作为出口网关连接到外网 R1，R2 和 R3 运行 VRRP 协议，两台路由器在同一个虚拟组。当 R2 为主路由器时，所有业务流量都由 R2 承担，高峰期时会造成网络阻塞，而 R3 一直处于空闲状态，这样就造成了一台路由器资源的浪费。现在为了优化公司网络，增加设备利用率，需要在 R2 和 R3 之间部署双备份组 VRRP，使得 R2、R3 分别为两个备份组的 Master，保证设备的利用率。

实验拓扑

配置 VRRP 多备份组的拓扑如图 9-2 所示。

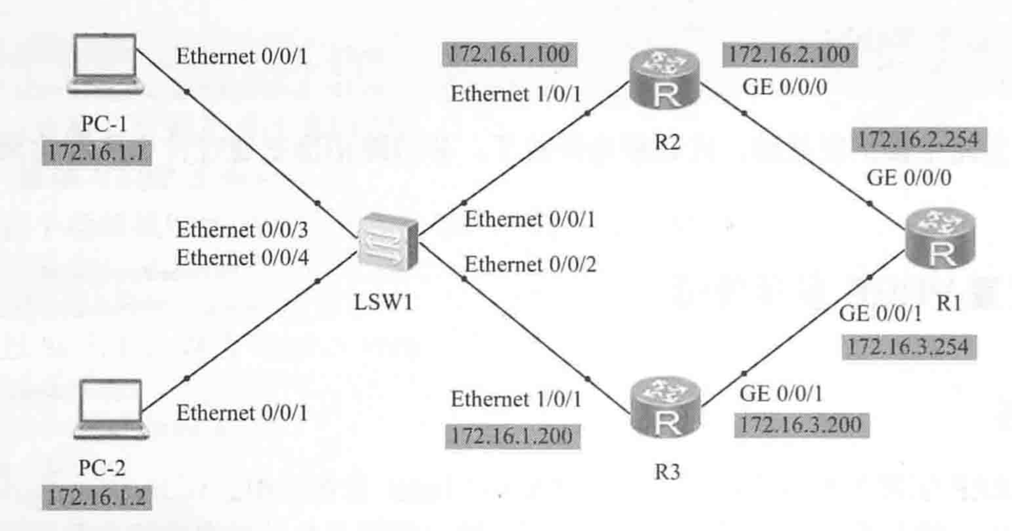


图 9-2 配置 VRRP 多备份组拓扑

实验编址

实验编址见表 9-2。

表 9-2 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR1220）	GE 0/0/0	172.16.2.254	255.255.255.0	N/A
	GE 0/0/1	172.16.3.254	255.255.255.0	N/A
R2（AR1220）	GE 0/0/0	172.16.2.100	255.255.255.0	N/A
	Ethernet 1/0/1	172.16.1.100	255.255.255.0	N/A
R3（AR1220）	GE 0/0/1	172.16.3.200	255.255.255.0	N/A
	Ethernet 1/0/1	172.16.1.200	255.255.255.0	N/A
PC-1	Ethernet 0/0/1	172.16.1.1	255.255.255.0	172.16.1.254
PC-2	Ethernet 0/0/1	172.16.1.2	255.255.255.0	172.16.1.253

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 地址配置，并使用 **ping** 命令检测各直连链路的连通性。

```
PC>ping 172.16.1.2
Ping 172.16.1.2: 32 data bytes, Press Ctrl_C to break
From 172.16.1.2: bytes=32 seq=1 ttl=128 time=31 ms
From 172.16.1.2: bytes=32 seq=2 ttl=128 time=32 ms
From 172.16.1.2: bytes=32 seq=3 ttl=128 time=31 ms
From 172.16.1.2: bytes=32 seq=4 ttl=128 time=15 ms
From 172.16.1.2: bytes=32 seq=5 ttl=128 time=16 ms
--- 172.16.1.2 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
0.00% packet loss
  round-trip min/avg/max = 15/25/32 ms
```

其余直连网段的连通性测试省略。

2. 部署 OSPF 网络

在公司的出口网关路由器 R2、R3 和外网路由器 R1 上配置 OSPF 协议，使用进程号 1，且所有网段均通告进区域 0 中。

```
[R1]ospf 1
[R1-ospf-1]area 0
[R1-ospf-1-area-0.0.0.0]network 172.16.2.0 0.0.0.255
[R1-ospf-1-area-0.0.0.0]network 172.16.3.0 0.0.0.255

[R2]ospf 1
[R2-ospf-1]area 0
[R2-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255
[R2-ospf-1-area-0.0.0.0]network 172.16.2.0 0.0.0.255

[R3]ospf 1
[R3-ospf-1]area 0
[R3-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 172.16.3.0 0.0.0.255
```

配置完成后，在 R1 上检查 OSPF 邻居建立情况。

```
[R1]display ospf peer brief
        OSPF Process 1 with Router-ID 172.16.2.254
                Peer Statistic Information
----------------------------------------------------------------------------
 Area Id          Interface                    Neighbor id       State
 0.0.0.0          GigabitEthernet0/0/0         172.16.1.100      Full
 0.0.0.0          GigabitEthernet0/0/1         172.16.1.200      Full
```

可以观察到，此时 R1 已经与 R2、R3 成功建立起了 OSPF 邻居关系。

3. 配置 VRRP 双备份组

为了提高网络的可靠性，公司采用双出口的形式连接到外网。但是如果采用普通的 VRRP 单备份组配置，就会有一台设备处在空闲状态。为了提高设备的利用率，网络管理员决定采用双备份组的配置，使得不同的设备成为不同备份组中的 Master，一起承担网络流量。

在 R2 和 R3 上创建 VRRP 虚拟组 1，虚拟 IP 为 172.16.1.254，指定 R2 的优先级为 120，R3 的优先级保持默认优先级不变。

```
[R2]interface Ethernet 1/0/1
[R2-Ethernet1/0/1]vrrp vrid 1 virtual-ip 172.16.1.254
[R2-Ethernet1/0/1]vrrp vrid 1 priority 120

[R3]interface Ethernet 1/0/1
[R3-Ethernet1/0/1]vrrp vrid 1 virtual-ip 172.16.1.254
```

配置完成后，分别查看 R2 和 R3 的 VRRP 信息。

```
[R2]display vrrp brief
Total:1      Master:1      Backup:0      Non-active:0
VRID  State        Interface              Type      Virtual IP
1     Master       Eth1/0/1               Normal    172.16.1.254

[R3]display vrrp brief
Total:1      Master:0      Backup:1      Non-active:0
VRID  State        Interface              Type      Virtual IP
1     Backup       Eth1/0/1               Normal    172.16.1.254
```

可以观察到，R2 为组 1 的 Master，R3 为 Backup。

在 R2 和 R3 上创建 VRRP 虚拟组 2，虚拟 IP 为 172.16.1.253，指定 R3 的优先级为 120，R2 的优先级保持默认优先级不变。

```
[R2]interface Ethernet 1/0/1
[R2-Ethernet1/0/1]vrrp vrid 2 virtual-ip 172.16.1.253

[R3]interface Ethernet 1/0/1
[R3-Ethernet1/0/1]vrrp vrid 2 virtual-ip 172.16.1.253
[R3-Ethernet1/0/1]vrrp vrid 2 priority 120
```

配置完成后，分别查看 R2 和 R3 的 VRRP 信息。

```
[R2]display vrrp brief
Total:2      Master:1      Backup:1      Non-active:0
VRID   State        Interface                Type       Virtual IP
1      Master       Eth1/0/1                 Normal     172.16.1.254
2      Backup       Eth1/0/1                 Normal     172.16.1.253

[R3]display vrrp brief
Total:2      Master:1      Backup:1      Non-active:0
VRID   State        Interface                Type       Virtual IP
1      Backup       Eth1/0/1                 Normal     172.16.1.254
2      Master       Eth1/0/1                 Normal     172.16.1.253
```

可以观察到，R3 为组 2 的 Master，R2 为 Backup。

在 PC-1 上设置网关地址为 172.16.1.254，PC-2 上设置网关地址为 172.16.1.253，并在 PC-1 上执行 **tracert** 172.16.2.254 命令，PC-2 上执行 **tracert** 172.16.3.254 命令。

```
PC>tracert 172.16.2.254
traceroute to 172.16.2.254, 8 hops max
(ICMP), press Ctrl+C to stop
 1  172.16.1.100   47 ms   15 ms   47 ms
 2  172.16.2.254   62 ms   63 ms   31 ms

PC>tracert 172.16.3.254
traceroute to 172.16.3.254, 8 hops max
(ICMP), press Ctrl+C to stop
 1  172.16.1.200   46 ms   47 ms   31 ms
 2  172.16.3.254   63 ms   62 ms   63 ms
```

观察发现 PC-1 现在是通过 R2 访问外网，PC-2 现在是通过 R3 访问外网，实现了网络优化的需求。

4. 验证 VRRP 抢占特性

在虚拟组 2 中 R3 为 Master 路由器，优先级为 120。现在虚拟组 2 中修改 R2 的抢占模式为非抢占方式（默认是抢占方式），并将优先级改为 200，即大于 R3 的优先级。

```
[R2]interface Ethernet 1/0/1
[R2-Ethernet1/0/1]vrrp vrid 2 preempt-mode disable
[R2-Ethernet1/0/1]vrrp vrid 2 priority 200
```

配置完成后，在 R2 上查看虚拟组 2 的信息。

```
[R2-Ethernet1/0/1]display vrrp
Ethernet1/0/1 | Virtual Router 1
State : Master
……
Ethernet1/0/1 | Virtual Router 2
```

```
State : Backup
Virtual IP : 172.16.1.253
Master IP : 172.16.1.200
PriorityRun : 200
PriorityConfig : 200
MasterPriority : 120
Preempt : NO
TimerRun : 1 s
......
```

可以观察到，尽管 R2 的配置优先级大于 R3，并且最终运行优先级也大于 R3，但是由于 R2 是非抢占模式，R2 不会抢占成为 Master。

5. 配置虚拟 IP 拥有者

在虚拟组 1 中，R2 的配置优先级为 120，R3 的配置优先级为默认的 100，R2 暂时是虚拟组 1 的 Master 路由器。现在网络管理员为了保证 R2 在虚拟组 1 始终是 Master，在 R2 的 E 1/0/1 接口上修改 IP 地址为 172.16.1.254/24，这样 R2 就成为了该虚拟组的虚拟 IP 地址拥有者。

```
[R2]interface Ethernet 1/0/1
[R2-Ethernet1/0/1]ip address 172.16.1.254 24
```

配置完成后，更改 R3 在虚拟组 1 的配置优先级为可配的最大值 254，这样 R3 的配置优先级就大于现在 R2 的配置优先级。

```
[R3]interface Ethernet 1/0/1
[R3-Ethernet1/0/1]vrrp vrid 1 priority 254
```

配置完成后，使用 **display vrrp brief** 命令查看主备状态。

```
[R3]display vrrp brief
VRID   State      Interface         Type     Virtual IP
1      Backup     Eth1/0/1          Normal   172.16.1.254
2      Master     Eth1/0/1          Normal   172.16.1.253
--------------------------------------------------------------------------
Total:2     Master:1     Backup:1     Non-active:0
```

观察发现 R3 无法抢占成为虚拟组 1 的 Master。

查看 R2 上的 VRRP 信息。

```
[R2]display vrrp
Ethernet1/0/1 | Virtual Router 1
State : Master
Virtual IP : 172.16.1.254
Master IP : 172.16.1.254
PriorityRun : 255
PriorityConfig : 120
MasterPriority : 255
Preempt : YES     Delay Time : 0 s
......
```

可以观察到，虽然 R2 在虚拟组 1 的配置优先级为 120，但是在成为了虚拟 IP 地址拥有者之后，其运行优先级为 255，高于 R3 的优先级 254，所以 R3 无法抢占成为该组的 Master。这再次验证了 Master 的选举及抢占都是比较运行优先级。

思考

在步骤 4 中，如果将 R2 配置成为虚拟组 2 的虚拟 IP 拥有者，试问此时 R2 能否抢

占成为 Master?

9.3 配置 VRRP 的跟踪接口及认证

原理概述

当 VRRP 的 Master 设备的上行接口出现问题，而 Master 设备一直保持 Active 状态，那么就会导致网络出现中断，所以必须要使得 VRRP 的运行状态和上行接口能够关联。在配置了 VRRP 冗余的网络中，为了进一步提高网络可靠性，需要在 Master 设备上配置上行接口监视，监视连接了外网的出接口。即当此接口断掉时，自动减小优先级一定的数值（该数值由人为配置），使减小后的优先级小于 Backup 设备的优先级，这样 Backup 设备就会抢占 Master 角色接替工作。

VRRP 支持报文的认证。默认情况下，设备对要发送和接收的 VRRP 报文不进行任何认证处理，认为收到的都是真实的、合法的 VRRP 报文。为了使 VRRP 运行更加安全和稳定，可以配置 VRRP 的认证。VRRP 支持简单字符（Simple）认证方式和 MD5 认证方式，用户可根据安全需要选择认证方式。

实验目的

- 理解 VRRP 监视接口的应用场景
- 掌握 VRRP 监视接口的配置方法
- 掌握 VRRP 认证的配置方法

实验内容

本实验模拟企业网络场景。该公司使用两台路由器 R2 和 R3 作为出口网关连接到外网路由器 R1，且 R2 和 R3 运行 VRRP 协议，两台路由器在同一个虚拟组 1，R2 为 Master 路由器。一次公司网络故障，突然所有主机都不能访问外网了，经检查发现是 R2 与外网路由器 R1 之间的链路断掉了，而 VRRP 的 Master 角色并没有发生切换，所有流量仍发送给 R2，导致无法访问外网。现在需对此网络进行优化，进一步提高可靠性和安全性，需要在 Master 设备上做 VRRP 的上行接口监视，当上行接口故障时，自动降低 VRRP 优先级使 Backup 设备能抢占 Master 角色，接替工作；当链路恢复时，R2 又能自动切换回 Master 设备，并且在 Master 与 Backup 设备之间配置 VRRP 认证，提高安全性。

实验拓扑

配置 VRRP 的接口监视及认证的拓扑如图 9-3 所示。

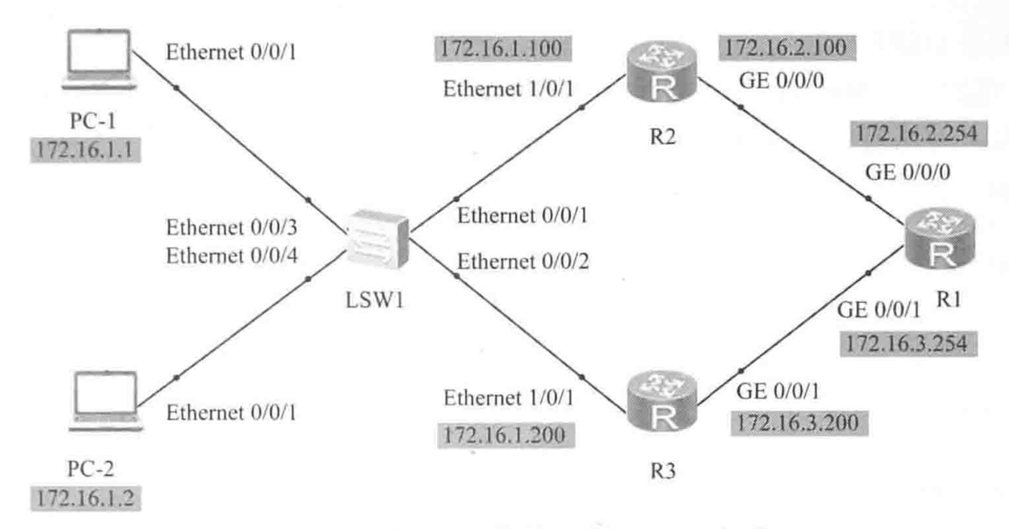


图 9-3 配置 VRRP 的接口监视及认证拓扑

实验编址

实验编址见表 9-3。

表 9-3 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR1220）	GE 0/0/0	172.16.2.254	255.255.255.0	N/A
	GE 0/0/1	172.16.3.254	255.255.255.0	N/A
R2（AR1220）	GE 0/0/0	172.16.2.100	255.255.255.0	N/A
	Ethernet 1/0/1	172.16.1.100	255.255.255.0	N/A
R3（AR1220）	GE 0/0/1	172.16.3.200	255.255.255.0	N/A
	Ethernet 1/0/1	172.16.1.200	255.255.255.0	N/A
PC-1	Ethernet 0/0/1	172.16.1.1	255.255.255.0	172.16.1.254
PC-2	Ethernet 0/0/1	172.16.1.2	255.255.255.0	172.16.1.254

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 地址配置，并使用 **ping** 命令检测各直连链路的连通性。

```
[R2]ping 172.16.1.1
  PING 172.16.1.1: 56  data bytes, press CTRL_C to break
    Reply from 172.16.1.1: bytes=56 Sequence=1 ttl=128 time=80 ms
    Reply from 172.16.1.1: bytes=56 Sequence=2 ttl=128 time=50 ms
    Reply from 172.16.1.1: bytes=56 Sequence=3 ttl=128 time=40 ms
    Reply from 172.16.1.1: bytes=56 Sequence=4 ttl=128 time=30 ms
    Reply from 172.16.1.1: bytes=56 Sequence=5 ttl=128 time=20 ms
  --- 172.16.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
0.00% packet loss
    round-trip min/avg/max = 20/44/80 ms
```

其余直连网段的连通性测试省略。

2. 部署 OSPF 网络

在公司的出口网关路由器 R2、R3 和外网路由器 R1 上配置 OSPF 协议，使用进程号 1，且所有网段均通告进区域 0 中。

```
[R1]ospf 1
[R1-ospf-1]area 0
[R1-ospf-1-area-0.0.0.0]network 172.16.2.0 0.0.0.255
[R1-ospf-1-area-0.0.0.0]network 172.16.3.0 0.0.0.255

[R2]ospf 1
[R2-ospf-1]area 0
[R2-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255
[R2-ospf-1-area-0.0.0.0]network 172.16.2.0 0.0.0.255

[R3]ospf 1
[R3-ospf-1]area 0
[R3-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 172.16.3.0 0.0.0.255
```

配置完成后，在 R1 上检查 OSPF 邻居建立情况。

```
[R1]display ospf peer brief
         OSPF Process 1 with Router-ID 172.16.2.254
                  Peer Statistic Information
----------------------------------------------------------------------------
 Area Id          Interface                    Neighbor id      State
 0.0.0.0          GigabitEthernet0/0/0         172.16.1.100     Full
 0.0.0.0          GigabitEthernet0/0/1         172.16.1.200     Full
```

可以观察到，此时 R1 已经与 R2、R3 成功建立起了 OSPF 邻居关系。

3. VRRP 基本配置

为了提高网络可靠性，公司采用双出口组网，并采用 VRRP 协议实现网关设备冗余。在 R2 与 R3 上创建同一个虚拟组，虚拟 ID 为 1，虚拟 IP 为 172.16.1.254，其中调整 R2 优先级为 120，使 R2 成为 Master，R3 上的优先级不变。

```
[R2]interface ethernet1/0/1
[R2-Ethernet1/0/1]vrrp vrid 1 virtual-ip 172.16.1.254
[R2-Ethernet1/0/1]vrrp vrid 1 priority 120

[R3]interface ethernet1/0/1
[R3-Ethernet1/0/1]vrrp vrid 1 virtual-ip 172.16.1.254
```

配置完成后，查看 R2、R3 上的 VRRP 信息。

```
[R2]display vrrp
  Ethernet1/0/1 | Virtual Router 1
  State : Master
  Virtual IP : 172.16.1.254
  Master IP : 172.16.1.100
  PriorityRun : 120
  PriorityConfig : 120
  MasterPriority : 120
  ......
```

```
[R3]display vrrp
  Ethernet1/0/1 | Virtual Router 1
  State : Backup
  Virtual IP : 172.16.1.254
  Master IP : 172.16.1.100
  PriorityRun : 100
  PriorityConfig : 100
  MasterPriority : 120
  ......
```

可以观察到 R2 现为 Master，R3 为 Backup。

此时公司网络发生故障，R2 与外网路由器 R1 之间的链路出现问题，无故断掉。关掉 R1 的 GE 0/0/0 接口模拟该故障。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]shutdown
```

配置完成后，查看主备的切换情况。

```
[R2]display vrrp
Ethernet1/0/1 | Virtual Router 1
State : Master
Virtual IP : 172.16.1.254
......
```

观察到路由器 R2 的 Master 角色并没有发生切换，所有流量仍发送给 R2，导致此时用户无法访问外网，连通性测试此处省略。即 VRRP 无法通过感知上行接口发生故障来完成主备设备切换。

4. 配置上行接口监视

为了进一步提高网络的可靠性和安全性，需要在 Master 设备 R2 上配置 VRRP 的上行接口监视。当 R2 的上行接口发生故障时，将自动降低优先级使得 Backup 设备能抢占 Master 角色，接替工作，将网络中断所造成的影响最小化。

在 R1 上恢复 GE 0/0/0 接口，并在 R2 上配置上行接口监视。监视上行接口 GE 0/0/0，当此接口断掉时，裁减优先级 50，使优先级变为 70，小于 R3 的优先级 100。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]undo shutdown

[R2]interface Ethernet1/0/1
[R2-Ethernet1/0/1]vrrp vrid 1 track interface GigabitEthernet 0/0/0 reduced 50
```

配置完成后，关闭 R1 的 GE 0/0/0 接口模拟故障发生，并使用 **display vrrp** 命令查看主备的切换情况。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]shutdown

[R2]display vrrp
Ethernet1/0/1 | Virtual Router 1
State : Backup
Virtual IP : 172.16.1.254
Master IP : 172.16.1.200
PriorityRun : 70
PriorityConfig : 120
```

```
MasterPriority : 100
Preempt : YES     Delay Time : 0 s
TimerRun : 1 s
TimerConfig : 1 s
Auth type : NONE
Virtual MAC : 0000-5e00-0101
Check TTL : YES
Config type : normal-vrrp
Backup-forward : disabled
Track IF : GigabitEthernet0/0/0     Priority reduced : 50
IF state : DOWN
Create time : 2013-06-23 18:53:52 UTC-05:13
Last change time : 2013-06-23 19:21:15 UTC-05:13

[R3]display vrrp
Ethernet1/0/1 | Virtual Router 1
State : Master
Virtual IP : 172.16.1.254
Master IP : 172.16.1.200
PriorityRun : 100
PriorityConfig : 100
MasterPriority : 100
Preempt : YES     Delay Time : 0 s
……
```

可以观察到，当 R2 上监视指定接口的状态为 DOWN 时，VRRP 优先级被裁减掉 50，变成 70，小于路由器 R3 的优先级 100，由于 R3 的 VRRP 默认为抢占模式，从而变成了 Backup，由 R3 成为新的 Master 并继续网络的转发。默认情况下，当被监视的接口变为 DOWN 时，VRRP 优先级的数值降低 10。

5. 在 R2 和 R3 上配置 VRRP 认证

默认情况下，设备对要发送的 VRRP 报文不进行任何认证处理，收到 VRRP 报文的设备也不进行任何检测，认为收到的都是真实的、合法的 VRRP 报文。可以通过配置更改 VRRP 的认证模式，使 VRRP 对报文进行验证，从而增强安全性。

在 R2 和 R3 上对 VRRP 虚拟组 1 配置接口认证，认证方式为 MD5，密码为 huawei。

```
[R2-Ethernet1/0/1]vrrp vrid 1 authentication-mode md5 huawei

[R3-Ethernet1/0/1]vrrp vrid 1 authentication-mode md5 huawei
```

注意在配置 VRRP 报文认证时，同一 VRRP 备份组的认证方式必须相同，否则 Master 设备和 Backup 设备无法协商成功。

配置完成后，使用 **display vrrp** 命令查看。

```
[R2]display vrrp
Ethernet1/0/1 | Virtual Router 1
……
Auth type : MD5     Auth key : %$%$!B56J6".AW`Os:5nOIM96GU"%$%$
Virtual MAC : 0000-5e00-0101
……

[R3]display vrrp
```

```
Ethernet1/0/1 | Virtual Router 1
......
Auth type : MD5      Auth key : %$%$xASELV]Z77V(rDFgUna@6FBd%$%$
Virtual MAC : 0000-5e00-0101
......
```

在以上显示信息中，可以观察到“Auth Type”字段显示为“MD5”，“Auth key”字段显示为“%$%$xASELV]Z77V(rDFgUna@6FBd%$%$”，即 VRRP 备份组 1 的报文认证方式为 MD5 认证，以密文为“%$%$xASELV]Z77V (rDFgUna@6FBd%$%$”形式显示。

思考

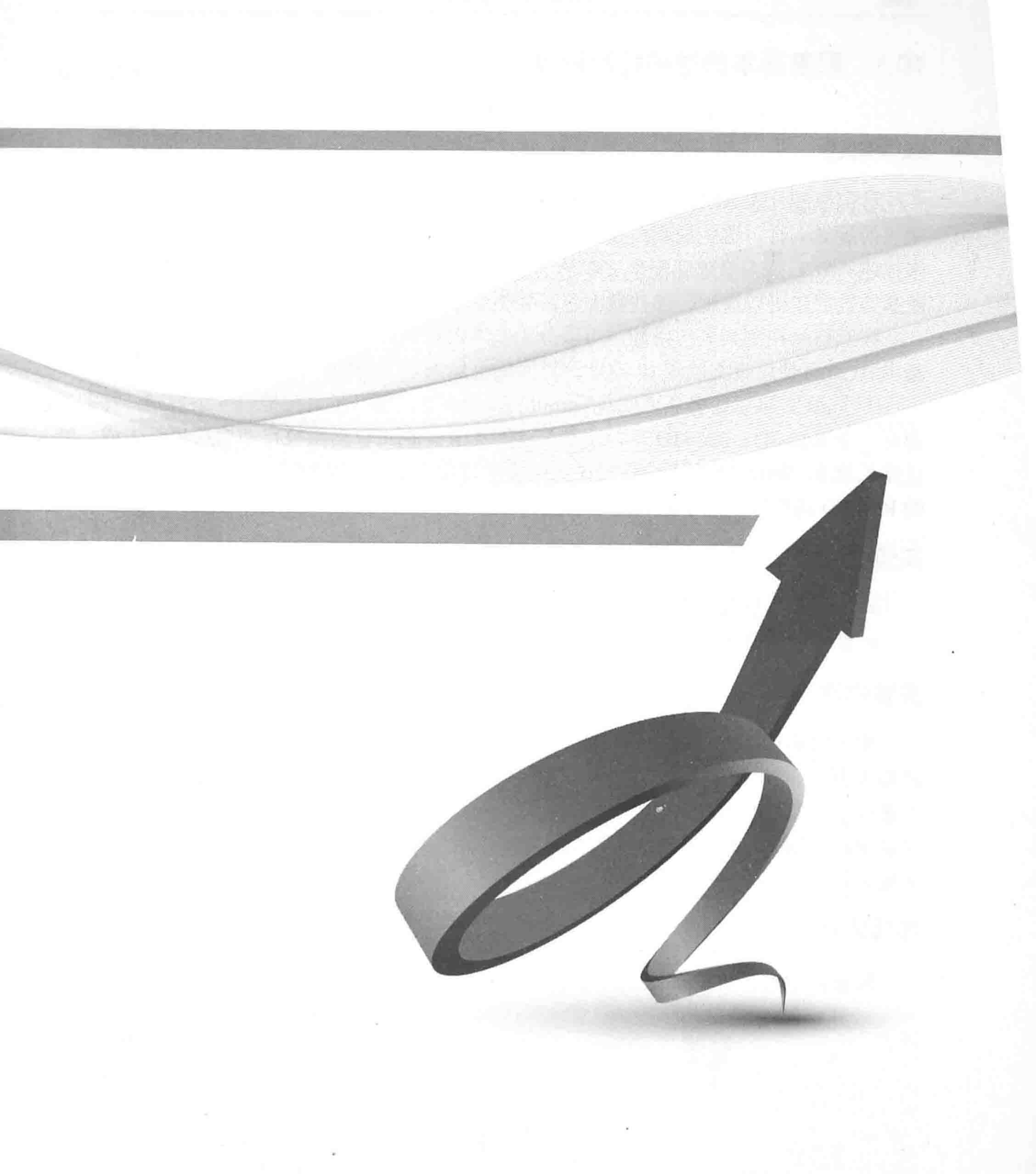
在本实验中，可以通过配置监视上行接口来提高 VRRP 的可靠性，但是监视上行接口仍然等同于监视直连链路，如果非直连链路出现故障，VRRP 协议能否感知？

第10章 基础过滤工具

10.1　配置基本的访问控制列表

10.2　配置高级的访问控制列表

10.3　配置前缀列表



10.1 配置基本的访问控制列表

原理概述

访问控制列表 ACL（Access Control List）是由 permit 或 deny 语句组成的一系列有顺序的规则集合，这些规则根据数据包的源地址、目的地址、源端口、目的端口等信息来描述。ACL 规则通过匹配报文中的信息对数据包进行分类，路由设备根据这些规则判断哪些数据包可以通过，哪些数据包需要拒绝。

按照访问控制列表的用途，可以分为基本的访问控制列表和高级的访问控制列表，基本 ACL 可使用报文的源 IP 地址、时间段信息来定义规则，编号范围为 2000～2999。

一个 ACL 可以由多条“deny/permit”语句组成，每一条语句描述一条规则，每条规则有一个 Rule-ID。Rule-ID 可以由用户进行配置，也可以由系统自动根据步长生成，默认步长为 5，Rule-ID 默认按照配置先后顺序分配 0、5、10、15 等，匹配顺序按照 ACL 的 Rule-ID 的顺序，从小到大进行匹配。

实验目的

- 理解基本访问控制列表的应用场景
- 掌握配置基本访问控制列表的方法

实验内容

本实验模拟企业网络环境，R1 为分支机构 A 管理员所在 IT 部门的网关，R2 为分支机构 A 用户部门的网关，R3 为分支机构 A 去往总部出口的网关设备，R4 为总部核心路由器设备。整网运行 OSPF 协议，并在区域 0 内。企业设计通过远程方式管理核心网路由器 R4，要求只能由 R1 所连的 PC（本实验使用环回接口模拟）访问 R4，其他设备均不能访问。

实验拓扑

配置基本的访问控制列表的拓扑如图 10-1 所示。

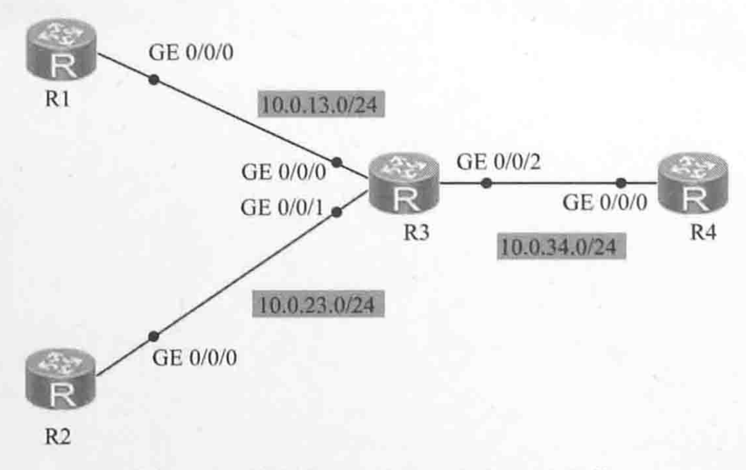


图 10-1 配置基本的访问控制列表拓扑

实验编址

实验编址见表 10-1。

表 10-1　实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR2220）	GE 0/0/0	10.0.13.1	255.255.255.0	N/A
	Loopback0	1.1.1.1	255.255.255.255	N/A
R2（AR2220）	GE 0/0/0	10.0.23.2	255.255.255.0	N/A
R3（AR2220）	GE 0/0/0	10.0.13.3	255.255.255.0	N/A
	GE 0/0/1	10.0.23.3	255.255.255.0	N/A
	GE 0/0/2	10.0.34.3	255.255.255.0	N/A
	Loopback0	3.3.3.3	255.255.255.255	N/A
R4（AR2220）	GE 0/0/0	10.0.34.4	255.255.255.0	N/A
	Loopback0	4.4.4.4	255.255.255.255	N/A

实验步骤

1. 基本配置

根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测各直连链路的连通性。

```
[R1]ping 10.0.13.3
PING 10.0.13.3: 56    data bytes, press CTRL_C to break
Reply from 10.0.13.3: bytes=56 Sequence=1 ttl=255 time=130 ms
Reply from 10.0.13.3: bytes=56 Sequence=2 ttl=255 time=60 ms
Reply from 10.0.13.3: bytes=56 Sequence=3 ttl=255 time=40 ms
Reply from 10.0.13.3: bytes=56 Sequence=4 ttl=255 time=30 ms
Reply from 10.0.13.3: bytes=56 Sequence=5 ttl=255 time=10 ms
--- 10.0.13.3 ping statistics ---
5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 10/54/130 ms
```

测试通过，其余直连网段的连通性测试省略。

2. 搭建 OSPF 网络

在所有路由器上运行 OSPF 协议，通告相应网段至区域 0 中。

```
[R1]ospf 1
[R1-ospf-1]area 0
[R1-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255
[R1-ospf-1-area-0.0.0.0]network 1.1.1.1 0.0.0.0

[R2]ospf 1
[R2-ospf-1]area 0
[R2-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255

[R3]ospf 1
[R3-ospf-1]area 0
[R3-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 10.0.34.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 3.3.3.3 0.0.0.0
```

```
[R4]ospf 1
[R4-ospf-1]area 0
[R4-ospf-1-area-0.0.0.0]network 10.0.34.0 0.0.0.255
[R4-ospf-1-area-0.0.0.0]network 4.4.4.4 0.0.0.0
```

配置完成之后，在 R1 的路由表上查看 OSPF 路由信息。

```
<R1>display ip routing-table protocol ospf
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Public routing table : OSPF
Destinations : 4          Routes : 4
OSPF routing table status : <Active>
Destinations : 4          Routes : 4
Destination/Mask    Proto   Pre  Cost    Flags  NextHop      Interface
3.3.3.3/32          OSPF    10   1       D      10.0.13.3    GigabitEthernet0/0/0
4.4.4.4/32          OSPF    10   2       D      10.0.13.3    GigabitEthernet0/0/0
10.0.23.0/24        OSPF    10   2       D      10.0.13.3    GigabitEthernet0/0/0
10.0.34.0/24        OSPF    10   2       D      10.0.13.3    GigabitEthernet0/0/0
OSPF routing table status : <Inactive>
Destinations : 0          Routes : 0
```

路由器 R1 已经学习到了相关网段的路由条目，测试 R1 的环回口与 R4 的环回口间的连通性。

```
<R1>ping -a 1.1.1.1 4.4.4.4
PING 4.4.4.4: 56   data bytes, press CTRL_C to break
Reply from 4.4.4.4: bytes=56 Sequence=1 ttl=254 time=20 ms
Reply from 4.4.4.4: bytes=56 Sequence=2 ttl=254 time=20 ms
Reply from 4.4.4.4: bytes=56 Sequence=3 ttl=254 time=10 ms
Reply from 4.4.4.4: bytes=56 Sequence=4 ttl=254 time=20 ms
Reply from 4.4.4.4: bytes=56 Sequence=5 ttl=254 time=20 ms
--- 4.4.4.4 ping statistics ---
5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 10/18/20 ms
```

通信正常，其他路由器之间测试省略。

3．配置基本 ACL 控制访问

在总部核心路由器 R4 上配置 Telnet 相关配置，配置用户密码为 huawei。

```
[R4]user-interface vty 0 4
[R4-ui-vty0-4]authentication-mode password
Please configure the login password (maximum length 16):huawei
```

配置完成后，尝试在 IT 部门网关设备 R1 上建立 Telnet 连接。

```
<R1>telnet 4.4.4.4
Press CTRL_ ] to quit telnet mode
Trying 4.4.4.4 ...
Connected to 4.4.4.4 ...
Login authentication
Password:
<R4>
```

可以观察到，R1 可以成功登录 R4。再尝试在普通员工部门网关设备 R2 上建立连接。

```
<R2>telnet 4.4.4.4
Press CTRL_ ] to quit telnet mode
Trying 4.4.4.4 ...
```

```
Connected to 4.4.4.4 ...
Login authentication
Password:
<R4>
```

这时发现，只要是路由可达的设备，并且拥有 Telnet 的密码，都可以成功访问核心设备 R4。这显然是极为不安全的。网络管理员通过配置标准 ACL 来实现访问过滤，禁止普通员工设备登录。

基本的 ACL 可以针对数据包的源 IP 地址进行过滤，在 R4 上使用 **acl** 命令创建一个编号型 ACL，基本 ACL 的范围是 2000~2999。

```
[R4]acl 2000
```

接下来在 ACL 视图中，使用 **rule** 命令配置 ACL 规则，指定规则 ID 为 5，允许数据包源地址为 1.1.1.1 的报文通过，反掩码为全 0，即精确匹配。

```
[R4-acl-basic-2000]rule 5 permit source 1.1.1.1 0
```

使用 **rule** 命令配置第二条规则，指定规则 ID 为 10，拒绝任意源地址的数据包通过。

```
[R4-acl-basic-2000]rule 10 deny source any
```

在上面的 ACL 配置中，第一条规则的规则 ID 定义为 5，并不是 1；第二条定义为 10，也不与 5 连续，这样配置的好处是能够方便后续的修改或插入新的条目。并且在配置的时候也可以不采用手工方式指定规则 ID，ACL 会自动分配规则 ID，第一条为 5，第二条为 10，第三条为 15，依此类推，即默认步长为 5，该步长参数也是可以修改的。

ACL 配置完成后，在 VTY 中调用。使用 inbound 参数，即在 R4 的数据入方向上调用。

```
[R4]user-interface vty 0 4
[R4-ui-vty0-4]acl 2000 inbound
```

配置完成后，使用 R1 的环回口地址 1.1.1.1 测试访问 4.4.4.4 的连通性。

```
<R1>telnet -a 1.1.1.1 4.4.4.4
Press CTRL_ ] to quit telnet mode
Trying 4.4.4.4 ...
Connected to 4.4.4.4 ...
Login authentication
Password:
<R4>
```

发现没有问题，然后尝试在 R2 上访问 R4。

```
<R2>telnet 4.4.4.4
Press CTRL_ ] to quit telnet mode
Trying 4.4.4.4 ...
Error: Can't connect to the remote host
<R2>
```

可以观察到，此时 R2 已经无法访问 4.4.4.4，即上述 ACL 配置已经生效。

4. 基本 ACL 的语法规则

ACL 的执行是有顺序性的，如果规则 ID 小的规则已经被命中，并且执行了允许或者拒绝的动作，那么后续的规则就不再继续匹配。

在 R4 上使用 **display acl all** 命令查看设备上所有的访问控制列表。

```
[R4]display acl all
```

```
 Total quantity of nonempty ACL number is 1
Basic ACL 2000, 2 rules
Acl's step is 5
rule 5 permit source 1.1.1.1 0
rule 10 deny
```

以上是目前 ACL 的所有配置信息。根据上一步骤中的配置，R4 中存在一个基本 ACL，有两个规则 rule 5 permit source 1.1.1.1 0 和 rule 10 deny source any，且根据这两个规则已经将 R2 的访问拒绝。现出现新的需求，需要 R3 能够使用其环回口 3.3.3.3 访问 R4。

首先尝试使用规则 ID 15 来添加允许 3.3.3.3 访问的规则。

```
[R4]acl 2000
[R4-acl-basic-2000]rule 15 permit source 3.3.3.3 0
```

配置完成后，尝试使用 R3 的 3.3.3.3 访问 R4。

```
<R3>telnet -a 3.3.3.3 4.4.4.4
Press CTRL_ ] to quit telnet mode
Trying 4.4.4.4 ...
Error: Can't connect to the remote host
<R3>
```

发现无法访问。按照 ACL 匹配顺序，这是由于规则为 10 的条目是拒绝所有行为，后续所有的允许规则都不会被匹配。若要此规则生效，必须添加在拒绝所有的规则 ID 之前。

在 R4 上修改 ACL 2000，将规则 ID 修改为 8。

```
[R4]acl 2000
[R4-acl-basic-2000]undo rule 15
[R4-acl-basic-2000]rule 8 permit source 3.3.3.3 0
```

配置完成后，再次尝试使用 R3 的环回口访问 R4。

```
<R3>telnet -a 3.3.3.3 4.4.4.4
Press CTRL_ ] to quit telnet mode
Trying 4.4.4.4 ...
Connected to 4.4.4.4 ...
Login authentication
Password:
<R4>
```

此时访问成功，证明配置已经生效。

思考

在本实验中，如果 ACL 不配置在 R4 上，那么该如何设置？有什么优缺点？

10.2 配置高级的访问控制列表

原理概述

基本的 ACL 只能用于匹配源 IP 地址，而在实际应用当中往往需要针对数据包的其他参数进行匹配，比如目的 IP 地址、协议号、端口号等，所以基本的 ACL 由于匹配的

局限性而无法实现更多的功能，所以就需要使用高级的访问控制列表。

高级的访问控制列表在匹配项上做了扩展，编号范围为 3000～3999，既可使用报文的源 IP 地址，也可使用目的地址、IP 优先级、IP 协议类型、ICMP 类型、TCP 源端口/目的端口、UDP 源端口/目的端口号等信息来定义规则。

高级访问控制列表可以定义比基本访问控制列表更准确、更丰富、更灵活的规则，也因此得到更加广泛的应用。

实验目的

- 理解高级访问控制列表的应用场景
- 掌握配置高级访问控制列表的方法
- 理解高级访问控制列表与基本访问控制列表的区别

实验内容

本实验模拟企业网络环境。R1 为分支机构 A 管理员所在 IT 部门的网关，R2 为分支机构 A 用户部门的网关，R3 为分支机构 A 去往总部出口的网关设备，R4 为总部核心路由器设备。企业原始设计思路想要通过远程方式管理核心网路由器 R4，要求由 R1 所连的 PC 可以访问 R4，其他设备均不能访问。同时又要求只能管理 R4 上的 4.4.4.4 这台服务器，另一台同样直连 R4 的服务器 40.40.40.40 不能被管理（本实验 PC 使用环回接口模拟）。

实验拓扑

配置高级的访问控制列表的拓扑如图 10-2 所示。

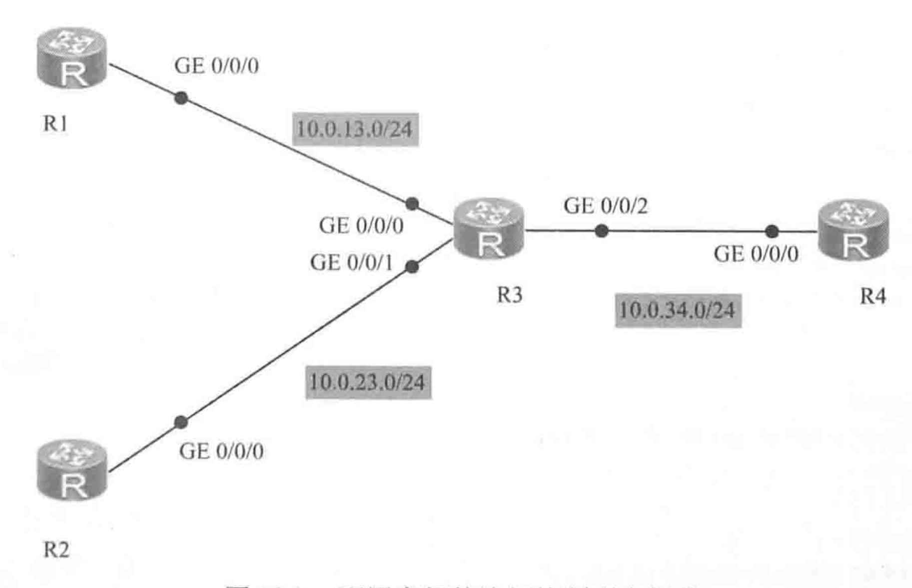


图 10-2　配置高级的访问控制列表拓扑

实验编址

实验编址见表 10-2。

表 10-2 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR2220）	GE 0/0/0	10.0.13.1	255.255.255.0	N/A
	Loopback 0	1.1.1.1	255.255.255.255	N/A
R2（AR2220）	GE 0/0/0	10.0.23.2	255.255.255.0	N/A
R3（AR2220）	GE 0/0/0	10.0.13.3	255.255.255.0	N/A
	GE 0/0/1	10.0.23.3	255.255.255.0	N/A
	GE 0/0/2	10.0.34.3	255.255.255.0	N/A
	Loopback 0	3.3.3.3	255.255.255.255	N/A
R4（AR2220）	GE 0/0/0	10.0.34.4	255.255.255.0	N/A
	Loopback 0	4.4.4.4	255.255.255.255	N/A
	Loopback 1	40.40.40.40	255.255.255.255	N/A

实验步骤

1. 基本配置

根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测各直连链路的连通性。

```
[R1]ping 10.0.13.3
PING 10.0.13.3: 56  data bytes, press CTRL_C to break
Reply from 10.0.13.3: bytes=56 Sequence=1 ttl=255 time=130 ms
Reply from 10.0.13.3: bytes=56 Sequence=2 ttl=255 time=60 ms
Reply from 10.0.13.3: bytes=56 Sequence=3 ttl=255 time=40 ms
Reply from 10.0.13.3: bytes=56 Sequence=4 ttl=255 time=30 ms
Reply from 10.0.13.3: bytes=56 Sequence=5 ttl=255 time=10 ms
--- 10.0.13.3 ping statistics ---
5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 10/54/130 ms
```

测试通过，其余直连网段的连通性测试省略。

2. 搭建 OSPF 网络

在所有路由器上运行 OSPF 协议，通告相应网段至区域 0 中。

```
[R1]ospf 1
[R1-ospf-1]area 0
[R1-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255
[R1-ospf-1-area-0.0.0.0]network 1.1.1.1 0.0.0.0

[R2]ospf 1
[R2-ospf-1]area 0
[R2-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255

[R3]ospf 1
[R3-ospf-1]area 0
[R3-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 10.0.34.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 3.3.3.3 0.0.0.0

[R4]ospf 1
[R4-ospf-1]area 0
```

```
[R4-ospf-1-area-0.0.0.0]network 10.0.34.0 0.0.0.255
[R4-ospf-1-area-0.0.0.0]network 4.4.4.4 0.0.0.0
[R4-ospf-1-area-0.0.0.0]network 40.40.40.40 0.0.0.0
```

配置完成之后，在 R1 的路由表上查看 OSPF 路由信息。

```
<R1>display ip routing-table protocol ospf
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Public routing table : OSPF
Destinations : 5          Routes : 5
OSPF routing table status : <Active>
Destinations : 5          Routes : 5
Destination/Mask    Proto   Pre  Cost   Flags   NextHop      Interface
3.3.3.3/32          OSPF    10   1      D       10.0.13.3    GigabitEthernet0/0/0
4.4.4.4/32          OSPF    10   2      D       10.0.13.3    GigabitEthernet0/0/0
40.40.40.40/32      OSPF    10   2      D       10.0.13.3    GigabitEthernet0/0/0
10.0.23.0/24        OSPF    10   2      D       10.0.13.3    GigabitEthernet0/0/0
10.0.34.0/24        OSPF    10   2      D       10.0.13.3    GigabitEthernet0/0/0
OSPF routing table status : <Inactive>
Destinations : 0          Routes : 0
```

路由器 R1 已经学习到了相关网段的路由条目。

3. 配置 Telenet

在总部核心路由器 R4 上配置 Telnet 相关配置，配置用户密码为 huawei。

```
[R4]user-interface vty 0 4
[R4-ui-vty0-4]authentication-mode password
Please configure the login password (maximum length 16):huawei
```

配置完成后，尝试在 R1 上建立与 R4 的环回接口 0 的 IP 地址的 Telnet 连接。

```
<R1>telnet –a 1.1.1.1 4.4.4.4
Press CTRL_ ] to quit telnet mode
Trying 4.4.4.4 ...
Connected to 4.4.4.4 ...
Login authentication
Password:
<R4>
```

可以观察到，R1 已经可以成功登录 R4。

再尝试在 R1 上建立与 R4 的环回接口 1 的 IP 地址的 Telnet 连接。

```
<R1>telnet –a 1.1.1.1 40.40.40.40
Press CTRL_ ] to quit telnet mode
Trying 40.40.40.40 ...
Connected to 40.40.40.40 ...
Login authentication
Password:
<R4>
```

这时发现，只要是路由可达的设备，并且拥有 Telnet 的密码，都可以成功正常登录。

4. 配置高级 ACL 控制访问

根据设计要求，R1 的环回接口只能通过 R4 上的 4.4.4.4 进行 Telnet 访问，但是不能通过 40.40.40.40 访问。

如果要 R1 只能通过访问 R4 的环回口 0 地址登录设备，即同时匹配数据包的源地址和目的地址实现过滤，此时通过标准 ACL 是无法实现的，因为 ACL 只能通过匹配源地址实现过滤，所以需要使用到高级 ACL。

在 R4 上使用 **acl** 命令创建一个高级 ACL 3000。

```
[R4]acl 3000
```

在高级 ACL 视图中，使用 **rule** 命令配置 ACL 规则，**ip** 为协议类型，允许源地址为 1.1.1.1、目的地址为 4.4.4.4 的数据包通过。

```
[R4-acl-adv-3000]rule    permit ip source 1.1.1.1 0 destination 4.4.4.4 0
```

配置完成后，查看 ACL 配置信息。

```
[R1-acl-adv-3000]dis acl all
 Total quantity of nonempty ACL number is 1
Advanced ACL 3000, 1 rule
Acl's step is 5
 rule 5 permit ip source 1.1.1.1 0 destination 4.4.4.4 0
```

可以观察到，在不指定规则 ID 的情况下，默认步长为 5，第一条规则的规则 ID 即为 5。

将 ACL 3000 调用在 VTY 下，使用 inbound 参数，即在 R4 的数据入方向上调用。

```
[R4]user-interface vty 0 4
[R4-ui-vty0-4]acl 3000 inbound
```

配置完成后，在 R1 上使用环回口地址尝试访问 40.40.40.40。

```
<R1>telnet -a 1.1.1.1 40.40.40.40
Press CTRL_ ] to quit telnet mode
Trying 40.40.40.40 ...
Error: Can't connect to the remote host
<R1>
```

可以观察到，此时过滤已经实现，R1 不能使用环回口地址访问 40.40.40.40。

此外高级 ACL 还可以实现对源、目的端口，协议号等信息的匹配，功能非常强大。

思考

路由器能否通过 ACL 过滤自身产生的数据包？

10.3 配置前缀列表

原理概述

前缀列表即 IP-Prefix List，它可以将与所定义的前缀列表相匹配的路由，根据定义的匹配模式进行过滤。前缀列表中的匹配条目由 IP 地址和掩码组成，IP 地址可以是网段地址或者主机地址，掩码长度的配置范围为 0～32，可以进行精确匹配或者在一定掩码长度范围内匹配，也可以通过配置关键字 greater-equal 和 less-equal 指定待匹配的前缀掩码长度范围。

前缀列表能同时匹配前缀号和前缀长度，主要用于路由的匹配和控制，不能用于数据包的过滤。

实验目的

- 理解前缀列表的应用场景
- 掌握前缀列表的配置方法

- 理解前缀列表与 ACL 的区别

实验内容

本实验模拟公司网络场景。公司分部 A 网络使用 11.1.1.0/24 网段，通过路由器 R2 和骨干路由器 R1 相连，网络运行 RIPv2 协议。现在公司新成立一个分部 B，新分部 B 的路由器 R3 连接 R1 加入该 RIPv2 网络。由于新分部 B 的网络管理员不熟悉公司内网 IP 地址规划，在新分部 B 中使用了 11.1.1.0/25 网段，这样导致从总部发往分部 A 的部分数据包在 R1 上都会由于路由掩码最长匹配从而错误地发往分部 B。而整个新分部 B 整改 IP 地址需要一定时间，公司当务之急是需要恢复总部与分部 A 的通信，可以通过在 R1 上使用前缀列表过滤掉这些错误的路由。

实验拓扑

配置前缀列表的拓扑如图 10-3 所示。

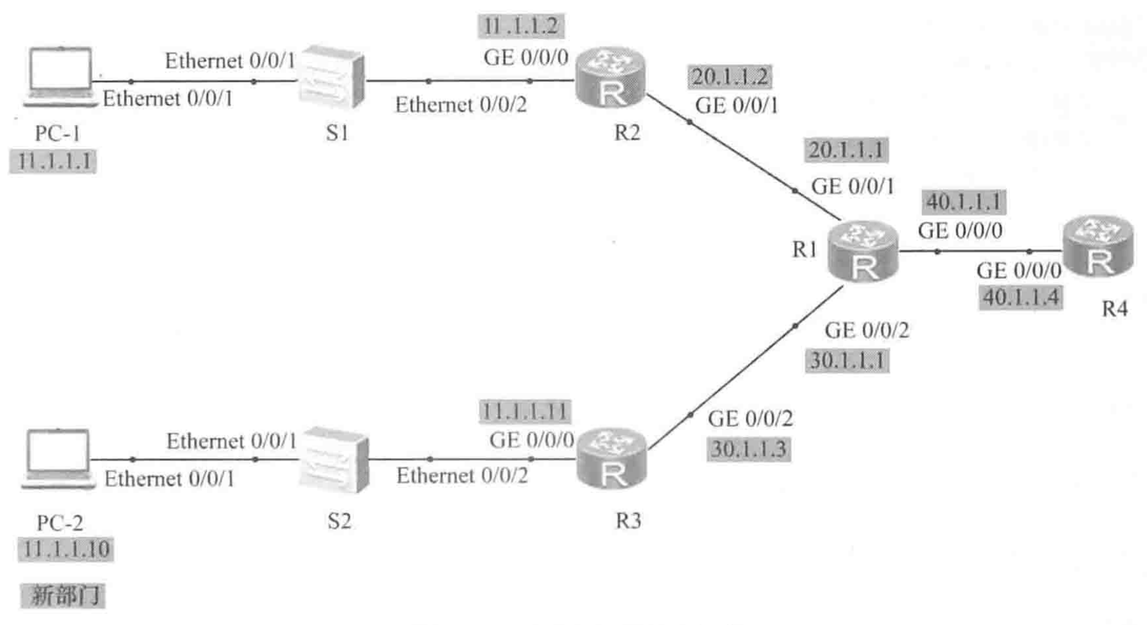


图 10-3　配置前缀列表拓扑

实验编址

实验编址见表 10-3。

表 10-3　　实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR2220）	GE 0/0/0	40.1.1.1	255.255.255.0	N/A
	GE 0/0/1	20.1.1.1	255.255.255.0	N/A
	GE 0/0/2	30.1.1.1	255.255.255.0	N/A
R2（AR2220）	GE 0/0/0	11.1.1.2	255.255.255.0	N/A
	GE 0/0/1	20.1.1.2	255.255.255.0	N/A
R3（AR2220）	GE 0/0/0	11.1.1.11	255.255.255.128	N/A
	GE 0/0/1	30.1.1.3	255.255.255.0	N/A

（续表）

设备	接口	IP 地址	子网掩码	默认网关
R4（AR2220）	GE 0/0/0	40.1.1.4	255.255.255.0	N/A
PC-1	Ethernet 0/0/1	11.1.1.1	255.255.255.0	11.1.1.2
PC-2	Ethernet 0/0/1	11.1.1.10	255.255.255.128	11.1.1.11

实验步骤

1. 基本配置

根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测各直连链路的连通性。

```
<R1>ping 20.1.1.2
  PING 20.1.1.2: 56    data bytes, press CTRL_C to break
    Reply from 20.1.1.2: bytes=56 Sequence=1 ttl=255 time=60 ms
    Reply from 20.1.1.2: bytes=56 Sequence=2 ttl=255 time=50 ms
    Reply from 20.1.1.2: bytes=56 Sequence=3 ttl=255 time=30 ms
    Reply from 20.1.1.2: bytes=56 Sequence=4 ttl=255 time=50 ms
Reply from 20.1.1.2: bytes=56 Sequence=5 ttl=255 time=40 ms
--- 20.1.1.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 1/38/110 ms
```

其他设备间的连通性测试略。

2. 搭建 RIP 网络

公司内部网络使用 RIPv2 协议。首先配置 R1、R2 和 R4 运行 RIPv2 协议，在总部路由器 R4 上能访问分部 A 的 PC。

```
[R1]rip 1
[R1-rip-1]version 2
[R1-rip-1]undo summary
[R1-rip-1]network 20.0.0.0
[R1-rip-1]network 30.0.0.0
[R1-rip-1]network 40.0.0.0

[R2]rip 1
[R2-rip-1]version 2
[R2-rip-1]undo summary
[R2-rip-1]network 11.0.0.0
[R2-rip-1]network 20.0.0.0

[R4]rip 1
[R4-rip-1]version 2
[R4-rip-1]undo summary
[R4-rip-1]network 40.0.0.0
```

配置完成后，查看总部 R4 的路由表。

```
[R4]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
```

```
Routing Tables: Public
         Destinations : 7          Routes : 7

Destination/Mask    Proto   Pre   Cost    Flags   NextHop        Interface
11.1.1.0/24         RIP     100   2       D       40.1.1.1       GigabitEthernet0/0/0
20.1.1.0/24         RIP     100   1       D       40.1.1.1       GigabitEthernet0/0/0
30.1.1.0/24         RIP     100   1       D       40.1.1.1       GigabitEthernet0/0/0
40.1.1.0/24         Direct  0     0       D       40.1.1.4       GigabitEthernet0/0/0
40.1.1.4/32         Direct  0     0       D       127.0.0.1      GigabitEthernet0/0/0
127.0.0.0/8         Direct  0     0       D       127.0.0.1      InLoopBack0
127.0.0.1/32        Direct  0     0       D       127.0.0.1      InLoopBack0
```

可以观察到，R4 的路由表中已经获得了 11.1.1.0/24 的路由，测试 R4 与公司总部 A 中 PC-1 间的连通性。

```
[R4]ping 11.1.1.1
  PING 11.1.1.1: 56  data bytes, press CTRL_C to break
    Reply from 11.1.1.1: bytes=56 Sequence=1 ttl=126 time=150 ms
    Reply from 11.1.1.1: bytes=56 Sequence=2 ttl=126 time=100 ms
    Reply from 11.1.1.1: bytes=56 Sequence=3 ttl=126 time=100 ms
    Reply from 11.1.1.1: bytes=56 Sequence=4 ttl=126 time=90 ms
    Reply from 10.1.1.1: bytes=56 Sequence=5 ttl=126 time=100 ms
……
```

现在新分部 B 加入公司网络，在 R3 上配置 RIPv2 协议。

```
[R3]rip 1
[R3-rip-1]version 2
[R3-rip-1]undo summary
[R3-rip-1]network 11.0.0.0
[R3-rip-1]network 30.0.0.0
```

配置完成后，再一次查看 R4 的路由表。

```
[R4]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8          Routes : 8
Destination/Mask    Proto   Pre   Cost    Flags   NextHop        Interface
11.1.1.0/24         RIP     100   2       D       40.1.1.1       GigabitEthernet0/0/0
11.1.1.0/25         RIP     100   2       D       40.1.1.1       GigabitEthernet0/0/0
20.1.1.0/24         RIP     100   1       D       40.1.1.1       GigabitEthernet0/0/0
30.1.1.0/24         RIP     100   1       D       40.1.1.1       GigabitEthernet0/0/0
40.1.1.0/24         Direct  0     0       D       40.1.1.4       GigabitEthernet0/0/0
40.1.1.4/32         Direct  0     0       D       127.0.0.1      GigabitEthernet0/0/0
127.0.0.0/8         Direct  0     0       D       127.0.0.1      InLoopBack0
127.0.0.1/32        Direct  0     0       D       127.0.0.1      InLoopBack0
```

可以观察到，此时 R4 接收到了公司分部 A 的 11.1.1.0/24 路由条目和新分部 B 的 11.1.1.0/25 路由条目，同样 R1 也会接收到这两条路由条目，这样会造成什么后果？

根据路由器转发数据的原理，在转发数据包时路由器会根据最长匹配的原则去匹配路由条目，即 R4 向分部 A 的终端 PC-1 发送数据时，当数据包到达 R1 后，根据包头的

目的 IP 地址与路由表中的路由条目进行匹配，发现 11.1.1.0/25 条目匹配更精确，这会使得数据包都根据这条路由条目进行转发，即将原本要发往 PC-1 的数据包都错误地发往 R3，造成总部与分部 A 异常通信。

在 R4 上测试与 PC-1 间的连通性。

```
<R4>ping 11.1.1.1
  PING 11.1.1.1: 56  data bytes, press CTRL_C to break
    Request time out
    Request time out
    Request time out
    Request time out
    Request time out
......
```

可以观察到此时无法正常通信。

在 R4 上使用 **tracert** 命令测试发往 PC-1 的数据包所经过的网关。

```
<R4>tracert 11.1.1.1
 traceroute to  11.1.1.1(11.1.1.1), max hops: 30 ,packet length: 40,press CTRL_C to break
 1 40.1.1.1 70 ms  50 ms  50 ms
 2 30.1.1.3 60 ms  80 ms  60 ms
 3  *  *  *
 4  *  *  *
```

可以观察到，此时 R4 发往分公司 A 的 PC-1 的数据包确实都已错误地发往 R3。

3. 配置 ACL 过滤路由

由于业务需要，现公司急需恢复总部与分部 A 间的通信。但是重新规划并配置整个分部 B 的 IP 地址需要一定时间，此时网络管理员尝试使用 ACL 来配置路由过滤，即在 R1 上过滤掉 11.1.1.0/25 这条路由。

在 R1 上创建基本的 ACL，拒绝 11.1.1.0 这个目的网段的路由。

```
[R1]acl number 2000
[R1-acl-basic-2000]rule 5 deny source 11.1.1.0 0.0.0.0
[R1-acl-basic-2000]rule 10 permit source any
```

接下来在 RIP 视图下，配置过滤策略（filter-policy），该策略通过调用之前配置好的 ACL 来达到过滤路由的目的，并且在 R1 的 RIP 路由进程中的接收方向应用此路由过滤策略。

```
[R1]rip 1
[R1-rip-1]filter-policy 2000 import
```

配置完成后，查看 R1 的路由表。

```
<R1>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 9        Routes : 9
Destination/Mask    Proto   Pre  Cost   Flags   NextHop      Interface
20.1.1.0/24         Direct  0    0      D       20.1.1.1     GigabitEthernet0/0/1
20.1.1.1/32         Direct  0    0      D       127.0.0.1    GigabitEthernet0/0/1
30.1.1.0/24         Direct  0    0      D       30.1.1.1     GigabitEthernet0/0/2
30.1.1.1/32         Direct  0    0      D       127.0.0.1    GigabitEthernet0/0/2
```

```
40.1.1.0/24     Direct  0    0    D    40.1.1.1     GigabitEthernet0/0/0
40.1.1.1/32     Direct  0    0    D    127.0.0.1    GigabitEthernet0/0/0
127.0.0.0/8     Direct  0    0    D    127.0.0.1    InLoopBack0
127.0.0.1/32    Direct  0    0    D    127.0.0.1    InLoopBack0
```

观察发现 R1 的路由表中 11.1.1.0/24 和 11.1.1.0/25 这两条路由都被过滤。这是因为 ACL 无法实现对掩码长度进行精确匹配，而分部 A 的网络位和分部 B 的网络位相同，都是 11.1.1.0，就会导致把分部 A 的路由也同时过滤。

4. 配置前缀列表过滤路由

为了能够精确匹配掩码长度，仅过滤掉 11.1.1.0/25 这条新分部 B 的路由，可以在 R1 上配置前缀列表。

在 R1 上配置前缀列表，同时精确匹配网络位和掩码长度。

```
[R1]ip ip-prefix 1 deny 11.1.1.0 25 greater-equal 25 less-equal 25
[R1]ip ip-prefix 1 permit 0.0.0.0 0 less-equal 32
```

第二条配置表示放行所有其他的路由，这是因为前缀列表也会有一条隐含的拒绝所有的规则，所以如果要放行其他所有路由的话，一定要显式增加一条允许所有的规则。而第一条配置也可以使用下面的方式简写。

```
[R1]ip ip-prefix 1 deny 11.1.1.0 25
```

将该前缀列表应用到过滤策略下。

```
[R1-rip-1]filter-policy ip-prefix 1 import
```

配置完成后，查看 R1 的路由表。

```
<R1>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 9        Routes : 9
Destination/Mask    Proto   Pre  Cost    Flags  NextHop      Interface
11.1.1.0/24         RIP     100  1       D      20.1.1.2     GigabitEthernet0/0/1
20.1.1.0/24         Direct  0    0       D      20.1.1.1     GigabitEthernet0/0/1
20.1.1.1/32         Direct  0    0       D      127.0.0.1    GigabitEthernet0/0/1
30.1.1.0/24         Direct  0    0       D      30.1.1.1     GigabitEthernet0/0/2
30.1.1.1/32         Direct  0    0       D      127.0.0.1    GigabitEthernet0/0/2
40.1.1.0/24         Direct  0    0       D      40.1.1.1     GigabitEthernet0/0/0
40.1.1.1/32         Direct  0    0       D      127.0.0.1    GigabitEthernet0/0/0
127.0.0.0/8         Direct  0    0       D      127.0.0.1    InLoopBack0
127.0.0.1/32        Direct  0    0       D      127.0.0.1    InLoopBack0
```

可以观察到，此时 R1 的路由表中仅存在 11.1.1.0/24，即分部 A 的路由条目，这样就恢复了总部与分部 A 的通信。测试总部与分部 A 中 PC-1 间的连通性。

```
<R4>ping 11.1.1.1
  PING 11.1.1.1: 56  data bytes, press CTRL_C to break
    Reply from 11.1.1.1: bytes=56 Sequence=1 ttl=126 time=120 ms
    Reply from 11.1.1.1: bytes=56 Sequence=2 ttl=126 time=50 ms
    Reply from 11.1.1.1: bytes=56 Sequence=3 ttl=126 time=130 ms
    Reply from 11.1.1.1: bytes=56 Sequence=4 ttl=126 time=60 ms
    Reply from 11.1.1.1: bytes=56 Sequence=5 ttl=126 time=110 ms
--- 11.1.1.1 ping statistics ---
    5 packet(s) transmitted
```

```
        5 packet(s) received
        0.00% packet loss
    round-trip min/avg/max = 1/38/110 ms
```

可以观察到，此时通信恢复正常。

5. 恢复新分部网络

为了新分部 B 能正确接入现有网络中，网络管理员需要重新规划 IP 地址，使分部 B 能与公司总部及分部 A 通信。

规划分部 B 网络使用 11.2.2.0/24 网段，更改 PC-2 的 IP 地址为 11.2.2.1/24，R3 的 G E 0/0/0 接口 IP 地址为 11.2.2.3/24。

```
[R3]interface GigabitEthernet 0/0/0
[R3-GigabitEthernet0/0/0]ip address 11.2.2.3
```

配置完成后，查看 R1 的路由表。

```
<R1>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 9          Routes : 9
Destination/Mask    Proto   Pre   Cost    Flags   NextHop      Interface
11.1.1.0/24         RIP     100   1       D       20.1.1.2     GigabitEthernet0/0/1
11.2.2.0/24         RIP     100   1       D       30.1.1.3     GigabitEthernet0/0/2
20.1.1.0/24         Direct  0     0       D       20.1.1.1     GigabitEthernet0/0/1
......
```

可以观察到，R1 的路由表中现有新分部 B 所在 10.2.2.0/24 网段的路由条目，也有分部 A 的路由。使用 **ping** 命令测试总部与分部 A、新分部 B 间的连通性。

```
<R4>ping 11.1.1.1
  PING 11.1.1.1: 56   data bytes, press CTRL_C to break
    Reply from 11.1.1.1: bytes=56 Sequence=1 ttl=126 time=110 ms
    Reply from 11.1.1.1: bytes=56 Sequence=2 ttl=126 time=70 ms
    Reply from 11.1.1.1: bytes=56 Sequence=3 ttl=126 time=70 ms
    Reply from 11.1.1.1: bytes=56 Sequence=4 ttl=126 time=80 ms
    Reply from 11.1.1.1: bytes=56 Sequence=5 ttl=126 time=100 ms
--- 11.1.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 1/38/110 ms

<R4>ping 11.2.2.1
  PING 11.2.2.1: 56   data bytes, press CTRL_C to break
    Reply from 11.2.2.1: bytes=56 Sequence=1 ttl=126 time=120 ms
    Reply from 11.2.2.1: bytes=56 Sequence=2 ttl=126 time=100 ms
    Reply from 11.2.2.1: bytes=56 Sequence=3 ttl=126 time=80 ms
    Reply from 11.2.2.1: bytes=56 Sequence=4 ttl=126 time=80 ms
    Reply from 11.2.2.1: bytes=56 Sequence=5 ttl=126 time=110 ms
--- 11.2.2.1 ping statistics ---
    5 packet(s) transmitted
```

```
        5 packet(s) received
        0.00% packet loss
round-trip min/avg/max = 1/38/110 ms
```

可以观察到，通信正常。

思考

如果将前缀列表中已配置好的语句顺序打乱会对实验结果产生影响吗？

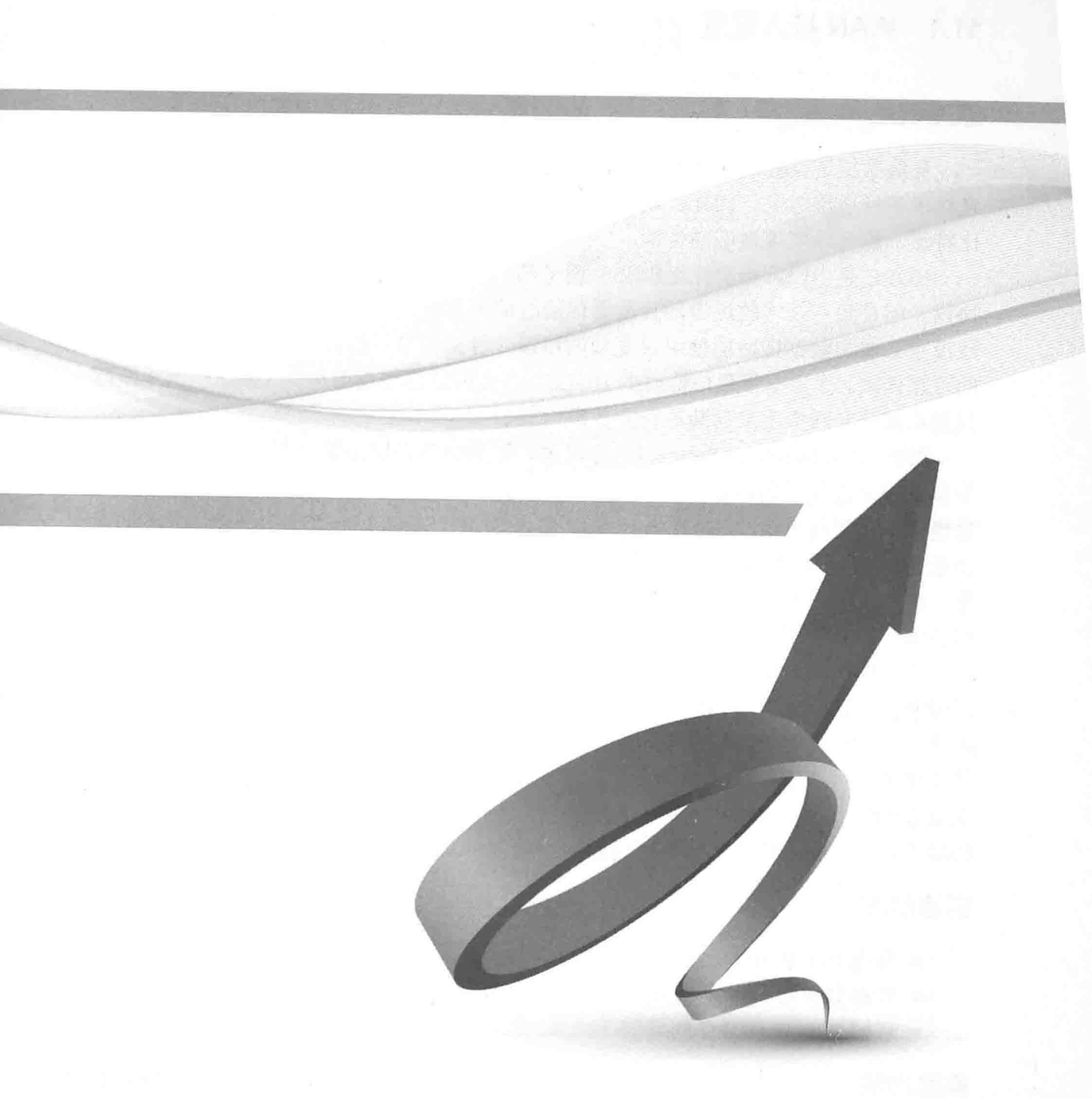
第11章 广域网

11.1 WAN接入配置

11.2 PPP的认证

11.3 帧中继基本配置

11.4 OSPF在帧中继网络中的配置



11.1 WAN 接入配置

原理概述

高级数据链路控制 HDLC（High-level Data Link Control）是一种链路层协议，运行在同步串行链路之上。HDLC 最大的特点是不需要规定数据必须是字符集，对任何一种比特流，均可以实现透明的传输。

HDLC 是由国际标准化组织 ISO 制定的，是通信领域曾广泛应用的一个数据链路层协议。但是随着技术的进步，目前通信信道的可靠性比过去已经有了非常大的改进，已经没有必要在数据链路层使用很复杂的协议（包括编号、检错重传等技术）来实现数据的可靠传输。作为窄带通信协议的 HDLC，在公网的应用逐渐消失，应用范围逐渐减小，只是在部分专网中用来封装透传业务数据。

PPP（Point-to-Point Protocol）协议是一种数据链路层协议，主要用来在全双工的同异步链路上进行点到点之间的数据传输。PPP 的设计初衷是为两个对等节点之间的 IP 流量提供一种封装协议，它是在串行线 IP 协议 SLIP（Serial Line IP）的基础上发展而来的。由于 SLIP 协议存在只支持异步传输方式、无协商过程、只能承载 IP 一种网络报文等问题，在发展过程中，逐步被 PPP 协议所替代。PPP 与 HDLC 的主要区别是：HDLC 是面向位的，而 PPP 是面向字节的。PPP 是一种多协议成帧机制，适用于调制解调器。

串行链路是指信息的各位数据被逐位按顺序传送的线路，适用于远距离通信，但速度较慢，与之相对的是并行链路，能够在同一时刻传送一个 8 bit 数据。同步和异步是广域网的串行链路的两种传输模式，同步模式要求通信双方以相同的时钟频率进行，通过共享单个时钟或定时脉冲源保证发送方和接收方的准确同步，效率较高；异步模式不要求双方同步，收发方可以采用各自的时钟源，双方遵循异步的通信协议，以字符为数据传输单位，发送方传送字符的时间间隔不确定，发送效率比同步模式低。

实验目的

- 掌握 PPP 的基本配置
- 掌握 HDLC 的基本配置
- 理解 PPP 与 HDLC 的异同

实验内容

本实验模拟企业网络场景。某公司开发部门的 PC-1，通过部门路由器 R2 连接到公司出口网关 R1；市场部门的 PC-2 直连到公司出口网关；IT 部门的 PC-3 通过部门路由器 R3 连接到公司出口网关。R2 与 R1 之间链路为串行链路，封装 PPP 协议；R3 与 R1 之间链路为串行链路，封装 HDLC 协议。R2 与 R3 分别设置默认路由指向 R1，使各部门之间能互相访问。

实验拓扑

WAN 接入配置的拓扑如图 11-1 所示。

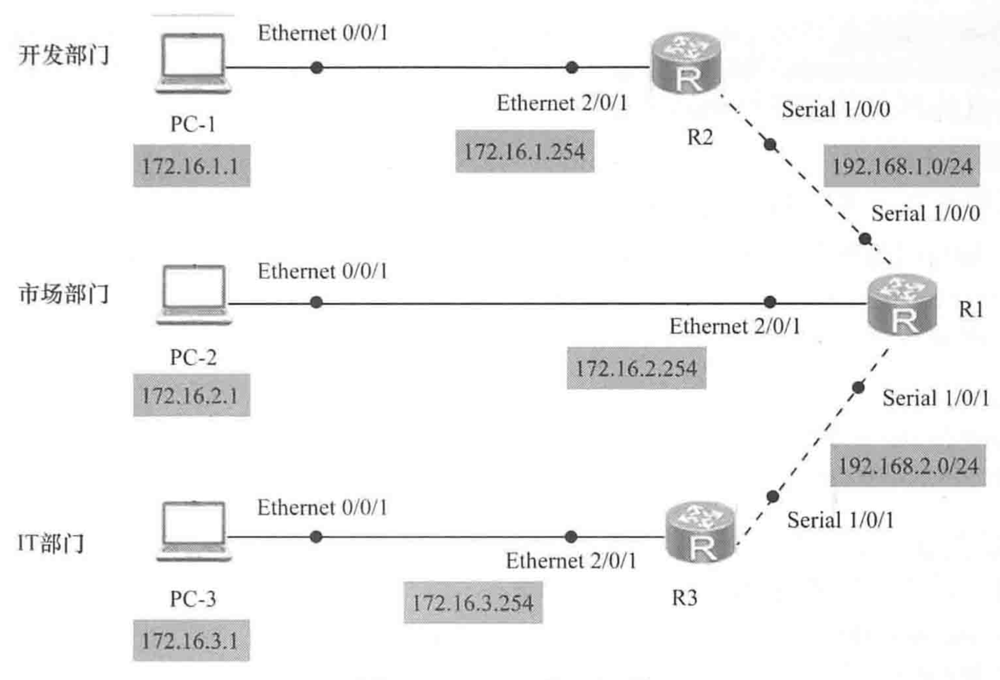


图 11-1　WAN 接入拓扑

实验编址

实验编址见表 11-1。

表 11-1　　实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR1220）	Serial 1/0/1	192.168.2.2	255.255.255.0	N/A
	Serial 1/0/0	192.168.1.2	255.255.255.0	N/A
	Ethernet 2/0/1	172.16.2.254	255.255.255.0	N/A
R2（AR1220）	Serial 1/0/0	192.168.1.1	255.255.255.0	192.168.1.2
	Ethernet 2/0/1	172.16.1.254	255.255.255.0	192.168.1.2
R3（AR1220）	Serial 1/0/1	192.168.2.1	255.255.255.0	192.168.2.2
	Ethernet 2/0/1	172.16.3.254	255.255.255.0	192.168.2.2
PC-1	Ethernet 0/0/1	172.16.1.1	255.255.255.0	172.16.1.254
PC-2	Ethernet 0/0/1	172.16.2.1	255.255.255.0	172.16.2.254
PC-3	Ethernet 0/0/1	172.16.3.1	255.255.255.0	172.16.3.254

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 地址配置，并使用 **ping** 命令检测各直连链路的连通性。

```
<R1>ping -c 1 172.16.2.1
  PING 172.16.2.1: 56  data bytes, press CTRL_C to break
    Reply from 172.16.2.1: bytes=56 Sequence=1 ttl=255 time=510 ms
  --- 172.16.2.1 ping statistics ---
    1 packet(s) transmitted
    1 packet(s) received
```

```
    0.00% packet loss
    round-trip min/avg/max = 510/510/510 ms
```

其余直连网段的连通性测试省略。

2. 配置 PPP

默认情况下，串行接口封装的链路层协议即为 PPP，可以直接在 R1 上使用 **display interface serial 1/0/0** 命令进行查看。

```
[R1]display interface serial 1/0/0
Serial1/0/0 current state : UP
……
Internet Address is 192.168.1.2/24
Link layer protocol is PPP
LCP opened, IPCP opened
……
```

在 R2 上配置默认路由指向出口网关 R1，并在 R1 上配置目的网段为 PC-1 所在网络的静态路由，下一跳路由器为 R2。

```
[R2]ip route-static 0.0.0.0 0.0.0.0 192.168.1.2
[R1]ip route-static 172.16.1.0 255.255.255.0 192.168.1.1
```

配置完成后，在 PC-1 上测试与 R1 间的连通性。

```
PC>ping 192.168.1.2
Ping 192.168.1.2: 32 data bytes, Press Ctrl_C to break
From 192.168.1.2: bytes=32 seq=1 ttl=254 time=16 ms
From 192.168.1.2: bytes=32 seq=2 ttl=254 time=31 ms
From 192.168.1.2: bytes=32 seq=3 ttl=254 time=32 ms
From 192.168.1.2: bytes=32 seq=4 ttl=254 time=31 ms
From 192.168.1.2: bytes=32 seq=5 ttl=254 time=31 ms
--- 192.168.1.2 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
0.00% packet loss
  round-trip min/avg/max = 16/28/32 ms
```

可以正常通信。

3. 配置 HDLC

在 R1 和 R3 的 S 1/0/1 接口上分别使用 **link-protocol** 命令配置链路层协议为 HDLC。

```
[R1]interface Serial 1/0/1
[R1-Serial1/0/1]link-protocol hdlc
Warning: The encapsulation protocol of the link will be changed.
Continue? [Y/N]:y

[R3]interface Serial 1/0/1
[R3-Serial1/0/1]link-protocol hdlc
Warning: The encapsulation protocol of the link will be changed.
Continue? [Y/N]:y
```

在 R3 上配置默认路由指向出口网关 R1，并在 R1 上配置目的网段为 PC-3 所在网络的静态路由，下一跳路由器为 R3 连接 R1 的 S 1/0/1 接口。

```
[R3]ip route-static 0.0.0.0 0.0.0.0 serial 1/0/1

[R1]ip route-static 172.16.3.0 255.255.255.0 serial 1/0/1
```

配置完成后，在 PC-3 上测试与路由器 R1 间的连通性。

```
PC>ping 192.168.2.2
```

```
Ping 192.168.2.2: 32 data bytes, Press Ctrl_C to break
From 192.168.2.2: bytes=32 seq=1 ttl=254 time=15 ms
From 192.168.2.2: bytes=32 seq=2 ttl=254 time=46 ms
From 192.168.2.2: bytes=32 seq=3 ttl=254 time=31 ms
From 192.168.2.2: bytes=32 seq=4 ttl=254 time=16 ms
From 192.168.2.2: bytes=32 seq=5 ttl=254 time=31 ms
--- 192.168.2.2 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
0.00% packet loss
  round-trip min/avg/max = 15/27/46 ms
```

可以正常通信。

思考

假设在某一直连链路上，一端接口的链路层协议为 PPP，另一端为 HDLC，此时能否正常通信？

11.2　PPP 的认证

原理概述

在网络日益发展的今天，人们对网络安全性的要求越来越高，而 PPP 协议之所以能成为广域网中应用较为广泛的协议，原因之一就是它能提供验证协议 CHAP（Challenge-Handshake Authentication Protocol，挑战式握手验证协议）、PAP（Password Authentication Protocol，密码验证协议），更好地保证了网络安全性。

PAP 为两次握手验证，口令为明文，验证过程仅在链路初始建立阶段进行。当链路建立阶段结束后，用户名和密码将由被验证方重复地在链路上发送给验证方，直到验证通过或者中止连接。PAP 不是一种安全的验证协议，因为口令是以明文方式在链路上发送的，并且用户名和口令还会被验证方不停地在链路上反复发送，导致很容易被截获。

CHAP 是三次握手验证协议，只在网络上传输用户名，而并不传输用户密码，因此安全性要比 PAP 高。CHAP 协议是在链路建立开始就完成的，在链路建立完成后的任何时间都可以进行再次验证。当链路建立阶段完成后，验证方发送一个“challenge”报文给被验证方；被验证方经过一次 Hash 算法后，给验证方返回一个值；验证方把自己经过 Hash 算法生成的值和被验证方返回的值进行比较。如果两者匹配，那么验证通过，否则验证不通过，连接被终止。

实验目的

- 掌握配置 PPP PAP 认证的方法
- 掌握配置 PPP CHAP 认证的方法
- 理解 PPP PAP 认证与 CHAP 认证的区别

实验内容

本实验模拟企业网络环境。R1 为分支机构接入端网关设备，PC-1 为企业分支机构终端。R2 为企业总部接入终端网关设备，PC-2 为企业总部终端，R3 为企业总部核心路由器。出于安全角度考虑，网络管理员在分支机构访问总部时部署 PPP 认证，R1 是被认证方路由器，R3 是认证方路由器，只有认证通过才能建立 PPP 连接进行正常访问。

实验拓扑

PPP 的认证拓扑如图 11-2 所示。

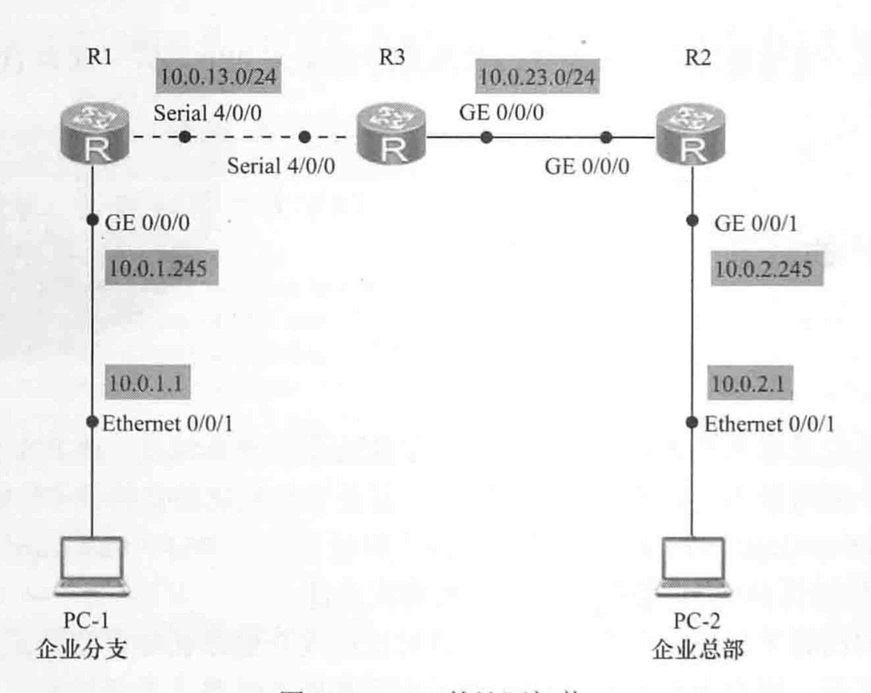


图 11-2 PPP 的认证拓扑

实验编址

实验编址见表 11-2。

表 11-2 实验编址

设备	接口	IP 地址	子网掩码	默认网关
PC-1	Ethernet 0/0/1	10.0.1.1	255.255.255.0	10.0.1.254
PC-2	Ethernet 0/0/1	10.0.2.1	255.255.255.0	10.0.2.254
R1（AR2220）	GE 0/0/0	10.0.1.254	255.255.255.0	N/A
	Serial 4/0/0	10.0.13.1	255.255.255.0	N/A
R2（AR2220）	GE 0/0/0	10.0.23.2	255.255.255.0	N/A
	GE 0/0/1	10.0.2.254	255.255.255.0	N/A
R3（AR2220）	GE 0/0/0	10.0.23.3	255.255.255.0	N/A
	Serial 4/0/0	10.0.13.3	255.255.255.0	N/A

实验步骤

1. 基本配置

根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测各直连链路的连通性。

```
<R1>ping 10.0.1.1
PING 10.0.1.1: 56  data bytes, press CTRL_C to break
Reply from 10.0.1.1: bytes=56 Sequence=1 ttl=255 time=90 ms
Reply from 10.0.1.1: bytes=56 Sequence=2 ttl=255 time=50 ms
Reply from 10.0.1.1: bytes=56 Sequence=3 ttl=255 time=1 ms
Reply from 10.0.1.1: bytes=56 Sequence=4 ttl=255 time=1 ms
Reply from 10.0.1.1: bytes=56 Sequence=5 ttl=255 time=1 ms
--- 10.0.1.1 ping statistics ---
5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 1/28/90 ms
```

测试通过，其余直连网段的连通性测试省略。

2. 搭建 OSPF 网络

在每台路由器上配置 OSPF 协议，并通告相应网段到区域 0 内。

```
[R1]ospf 1
[R1-ospf-1]area 0
[R1-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255
[R1-ospf-1-area-0.0.0.0]network 10.0.1.0 0.0.0.255

[R2]ospf 1
[R2-ospf-1]area 0
[R2-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255
[R2-ospf-1-area-0.0.0.0]network 10.0.2.0 0.0.0.255

[R3]ospf 1
[R3-ospf-1]area 0
[R3-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255
[R3-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255
```

配置完成后测试总部与分支终端间的连通性。

```
PC>ping 10.0.2.1
Ping 10.0.2.1: 32 data bytes, Press Ctrl_C to break
From 10.0.2.1: bytes=32 seq=1 ttl=252 time=78 ms
From 10.0.2.1: bytes=32 seq=2 ttl=252 time=78 ms
From 10.0.2.1: bytes=32 seq=3 ttl=252 time=63 ms
From 10.0.2.1: bytes=32 seq=4 ttl=252 time=31 ms
From 10.0.2.1: bytes=32 seq=5 ttl=252 time=31 ms
--- 10.0.2.1 ping statistics ---
5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 31/56/78 ms
```

可以观察到，通信正常。

3. 配置 PPP 的 PAP 认证

现在为了提升分支机构与总部通信时的安全性，在分支网关设备 R1 与公司核心设

备 R3 上部署 PPP 的 PAP 认证。R3 作为认证路由器，R1 作为被认证路由器。

由于在华为路由器上，广域网串行接口默认链路层协议即为 PPP，因此可以直接配置 PPP 认证。在总部设备 R3 上使用 **ppp authentication-mode** 命令设置本端的 PPP 协议对对端设备的认证方式为 PAP，认证采用的域名为 huawei。

```
[R3]interface Serial 4/0/0
[R3-Serial4/0/0]ppp authentication-mode pap domain huawei
```

接下来配置认证路由器 R3 的本地认证信息。

执行 **aaa** 命令，进入 AAA 视图。

```
[R3]aaa
```

使用 **authentication-scheme** 命令创建认证方案 huawei_1，并进入认证方案视图。

```
[R3-aaa]authentication-scheme huawei_1
```

使用 **authentication-mode** 命令配置认证模式为本地认证。

```
[R3-aaa-authen-huawei_1]authentication-mode local
```

使用 **domain** 命令创建域 huaweiyu，并进入域视图。

```
[R3-aaa]domain huaweiyu
```

使用 **authentication-scheme** 命令配置域的认证方案为 huawei_1，注意必须和创建的认证方案一致。

```
[R3-aaa-domain-huaweiyu]authentication-scheme huawei_1
```

退回到 AAA 视图，使用 **local-user** 命令配置存储在本地，为对端认证方所使用的用户名为 R1@huaweiyu，密码为 Huawei。

```
[R3-aaa]local-user R1@huaweiyu password cipher Huawei
[R3-aaa]local-user R1@huaweiyu service-type ppp
```

配置完成后，关闭 R1 与 R3 相连接口一段时间后再打开，使 R1 与 R3 间的链路重新协商，并检查链路状态和连通性。

```
[R3]interface Serial 4/0/0
[R3-Serial4/0/0]shutdown
[R3-Serial4/0/0]undo shutdown

<R1>display ip interface brief
*down: administratively down
......
Interface                    IP Address/Mask      Physical     Protocol
GigabitEthernet0/0/0         10.0.1.254/24        up           up
GigabitEthernet0/0/1         unassigned           down         down
GigabitEthernet0/0/2         unassigned           down         down
NULL0                        unassigned           up           up(s)
Serial4/0/0                  10.0.13.1/24         up           down
Serial4/0/1                  unassigned           down         down

<R3>ping 10.0.13.1
PING 10.0.13.1: 56   data bytes, press CTRL_C to break
Request time out
Request time out
Request time out
Request time out
Request time out
......
```

可以观察到，现在 R1 与 R3 间无法正常通信，链路物理状态正常，但是链路层协议

状态不正常。这是因为此时 PPP 链路上的 PAP 认证未通过，现在仅仅配置了被认证方设备 R3，还需要在认证方 R1 上配置相关 PAP 认证参数。

在 R1 上的 S 4/0/0 接口下，使用 **ppp pap local-user** 命令配置本端被对端以 PAP 方式验证时本地发送的 PAP 用户名和密码。

```
[R1]interface Serial 4/0/0
[R1-Serial4/0/0]ppp pap local-user R1@huaweiyu password cipher Huawei
```

配置完成后，再次查看链路状态并测试连通性。

```
[R1]display ip interface brief
*down: administratively down
……
Interface                    IP Address/Mask      Physical    Protocol
GigabitEthernet0/0/0         10.0.1.254/24        up          up
GigabitEthernet0/0/1         unassigned           down        down
GigabitEthernet0/0/2         unassigned           down        down
NULL0                        unassigned           up          up(s)
Serial4/0/0                  10.0.13.1/24         up          up
Serial4/0/1                  unassigned           down        down

<R1>ping 10.0.13.3
PING 10.0.13.3: 56  data bytes, press CTRL_C to break
Reply from 10.0.13.3: bytes=56 Sequence=1 ttl=255 time=70 ms
Reply from 10.0.13.3: bytes=56 Sequence=2 ttl=255 time=30 ms
Reply from 10.0.13.3: bytes=56 Sequence=3 ttl=255 time=50 ms
Reply from 10.0.13.3: bytes=56 Sequence=4 ttl=255 time=40 ms
Reply from 10.0.13.3: bytes=56 Sequence=5 ttl=255 time=10 ms
--- 10.0.13.3 ping statistics ---
5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 10/40/70 ms
```

可以观察到，现在 R1 与 R3 间的链路层协议状态正常，并且可以正常通信。

测试 PC-1 与 PC-2 的连通性。

```
PC>ping 10.0.2.1
Ping 10.0.2.1: 32 data bytes, Press Ctrl_C to break
From 10.0.2.1: bytes=32 seq=1 ttl=125 time=31 ms
From 10.0.2.1: bytes=32 seq=2 ttl=125 time=63 ms
From 10.0.2.1: bytes=32 seq=3 ttl=125 time=47 ms
From 10.0.2.1: bytes=32 seq=4 ttl=125 time=31 ms
From 10.0.2.1: bytes=32 seq=5 ttl=125 time=31 ms
--- 10.0.2.1 ping statistics ---
5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 31/40/63 ms
```

总部与分支间的终端通信正常。

4. 配置 PPP 的 CHAP 认证

公司网络管理员在日常网络维护中发现，分部公司频繁遭受攻击，PPP 认证密码经常被盗用，遂对网络状况进行分析。抓取 R1 的 S 4/0/0 数据包进行分析，如图 11-3 所示。

```
PPP Password Authentication Protocol
  Code: Authenticate-Request (0x01)
  Identifier: 0x01
  Length: 23
⊟ Data (19 bytes)
  ⊟ Peer ID length: 11 bytes
      Peer-ID (11 bytes)
  ⊟ Password length: 6 bytes
      Password (6 bytes)
```

图 11-3 抓包观察

可以观察到，在数据包中很容易找到所配置的用户名和密码。“Peer-ID”显示内容为用户名，“Password”显示内容为密码，可以查看具体内容，如图 11-4 所示。



图 11-4 抓包观察

很容易找到用户名为 R1@huaweiyu，密码为 Huawei。由此验证了使用 PAP 认证时，口令将以明文方式在链路上传送，并且由于完成 PPP 链路建立后，被认证方会不停地在链路上反复发送用户名和口令，直到身份认证过程结束，所以不能防止攻击。而使用 CHAP 认证时，口令用 MD5 算法加密后在链路上发送，能有效地防止攻击。为了进一步提高链路安全性，网络管理员需要重新部署 PPP 的 CHAP 认证。

首先删除原有的 PAP 认证配置，域名保持不变。

```
[R3]interface Serial 4/0/0
[R3-Serial4/0/0]undo ppp authentication-mode

[R1]interface Serial 4/0/0
[R1-Serial4/0/0]undo ppp pap local-user
```

删除后，在认证设备 R3 的 S 4/0/0 接口下配置 PPP 的认证方式为 CHAP。

```
[R3]interface Serial 4/0/0
[R3-Serial4/0/0]ppp authentication-mode CHAP
```

配置存储在本地，对端认证方所使用的用户名为 R1，密码为 huawei。

```
[R3]aaa
[R3-aaa]local-user R1 password cipher huawei
[R3-aaa]local-user R1 service-type ppp
```

其余认证方案和域的配置保持不变。

配置完成后，关闭 R1 与 R3 相连接口一段时间后再打开，使链路重新协商。查看链路状态，并测试连通性。

```
[R3]interface Serial4/0/0
[R3-Serial4/0/0]shutdown
[R3-Serial4/0/0]undo shutdown

<R3>display ip interface brief
*down: administratively down
......
Interface                    IP Address/Mask      Physical    Protocol
GigabitEthernet0/0/0         10.0.23.3/24         up          up
GigabitEthernet0/0/1         unassigned           down        down
GigabitEthernet0/0/2         unassigned           down        down
NULL0                        unassigned           up          up(s)
Serial4/0/0                  10.0.13.3/24         up          down
```

```
Serial4/0/1                    unassigned          down          down

<R3>ping 10.0.13.1
PING 10.0.13.1: 56  data bytes, press CTRL_C to break
Request time out
Request time out
Request time out
Request time out
Request time out
--- 10.0.13.1 ping statistics ---
5 packet(s) transmitted
0 packet(s) received
100.00% packet loss
```

可以观察到，目前 R1 与 R3 间的链路层协议状态不正常，无法正常通信。这是由于此时被认证方 R1 上还没有配置用户名和密码。

在 R1 的 S 4/0/0 接口下配置 CHAP 认证的用户名和密码。

```
[R1]interface Serial 4/0/0
[R1-Serial4/0/0]ppp chap user R1
[R1-Serial4/0/0]ppp chap password huawei
```

配置完成，测试 R1 与 R3 的连通性。

```
<R1>ping 10.0.13.3
PING 10.0.13.3: 56  data bytes, press CTRL_C to break
Reply from 10.0.13.3: bytes=56 Sequence=1 ttl=255 time=50 ms
Reply from 10.0.13.3: bytes=56 Sequence=2 ttl=255 time=30 ms
Reply from 10.0.13.3: bytes=56 Sequence=3 ttl=255 time=40 ms
Reply from 10.0.13.3: bytes=56 Sequence=4 ttl=255 time=40 ms
Reply from 10.0.13.3: bytes=56 Sequence=5 ttl=255 time=10 ms
--- 10.0.13.3 ping statistics ---
5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 10/30/50 ms
```

认证通过，R1 与 R3 间通信正常，再测试分支 PC-1 和总部 PC-2 间的连通性。

```
PC>ping 10.0.2.1
Ping 10.0.2.1: 32 data bytes, Press Ctrl_C to break
From 10.0.2.1: bytes=32 seq=1 ttl=125 time=31 ms
From 10.0.2.1: bytes=32 seq=2 ttl=125 time=32 ms
From 10.0.2.1: bytes=32 seq=3 ttl=125 time=47 ms
From 10.0.2.1: bytes=32 seq=4 ttl=125 time=46 ms
From 10.0.2.1: bytes=32 seq=5 ttl=125 time=32 ms
--- 10.0.2.1 ping statistics ---
5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 31/37/47 ms
```

在 R1 的 S 4/0/0 接口下再次抓取数据包查看，如图 11-5 所示。

```
⊟ PPP Challenge Handshake Authentication Protocol
    Code: Response (2)
    Identifier: 1
    Length: 23
  ⊟ Data (19 bytes)
      Value Size: 16
      Value: 1fa05959a98fe6e52edf581aa49ebc45
      Name: R1
```

图 11-5　抓包观察

可以观察到，现在数据包内容已经为加密方式发送，无法被攻击者截获认证密码，安全性得到了提升。

思考

当 PPP 链路 UP 后，在 PPP 链路一端加上认证配置而另一端不加，为什么一定要重启端口后认证才能生效?

11.3 帧中继基本配置

原理概述

帧中继（Frame Relay）是一种面向连接的数据链路层技术，主要用在公共或专用网上的局域网互联以及广域网连接。

帧中继协议是一种简化 X.25 的广域网协议，它在控制层面上提供虚电路的管理、带宽管理和防止阻塞等功能。在传送数据时使用的传输链路是逻辑连接，而不是物理连接。在一个物理连接上可以复用多个逻辑连接，实现带宽的复用和动态分配，帧透明传输和错误检测，但不提供重传操作。与传统的电路交换相比，帧中继网络有利于多用户、多速率数据的传输，也充分利用了网络资源。

帧中继网络两端的设备用虚电路来连接。每条虚电路是用数据链路连接标识符定义的一条帧中继连接通道，提供了用户设备（如路由器和主机等）之间进行数据通信的能力。帧中继网络的相关术语如下：

■ DTE（Data Terminal Equipment，数据终端设备）：通常指用户侧的主机或终端等；

■ DCE（Data Circuit-terminating Equipment，数据电路终结设备）：为用户设备提供接入的设备，属于网络设备，如帧中继交换机；

■ DLCI（Data Link Connection Identifier，数据链路连接标识）：虚链路接口的标识。帧中继能够在单一物理传输线路上提供多条虚电路，虚电路便通过 DLCI 来区分；

■ PVC（Permanent Virtual Circuit，永久虚电路）：永久虚电路是指给用户提供固定的虚电路，该电路一旦建立，则链路永远生效，除非管理员手动删除。PVC 用于两端之间频繁的、流量稳定的数据传输。

逆向地址解析协议（Inverse ARP）的主要功能是求解每条虚电路连接的对端设备的 IP 地址。如果知道了某条虚电路连接的对端设备的 IP 协议地址，在本地就可以生成对端 IP 地址与 DLCI 的映射（MAP），从而避免手工配置地址映射。

实验目的

- 掌握帧中继交换机的配置
- 掌握动态映射的配置
- 掌握静态映射的配置
- 掌握子接口和 DLCI 的映射配置

实验内容

本实验模拟企业网络场景。公司 A 的总部和分部分别设在不同地方，总部路由器 R1 和分部路由器 R2 通过帧中继网络相连，总部与分部之间申请了一条 PVC。由于业务的发展，公司 A 与公司 B 有了密切的业务来往，公司 B 路由器 R3 也采用帧中继并使用动态映射方式与公司 A 相连，即只能与公司 A 总部直接通信。现需要采用帧中继子接口配置和静态路由使 R3 能通过 R1 访问 R2，实现全网全通。

实验拓扑

帧中继基本配置的拓扑如图 11-6 所示。

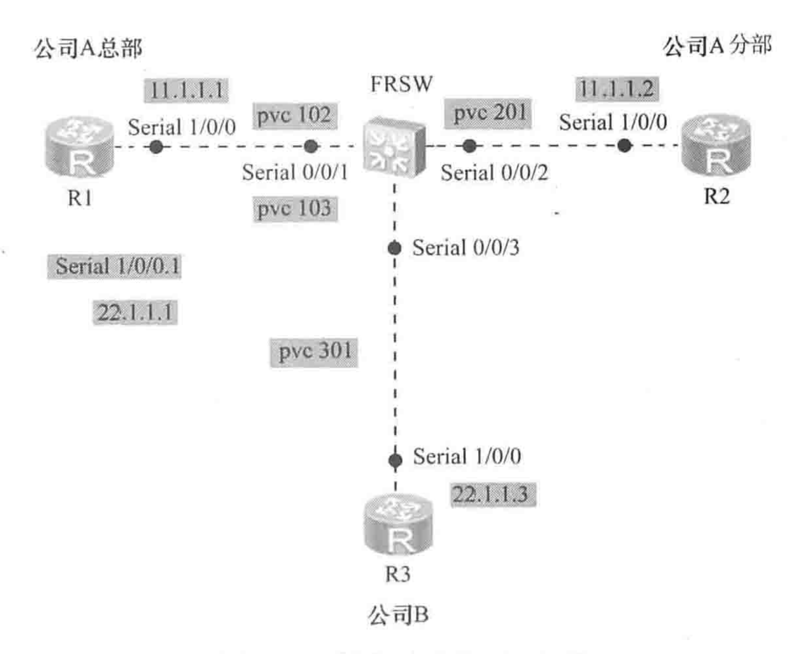


图 11-6　帧中继基本配置拓扑

实验编址

实验编址见表 11-3。

表 11-3　　实验编址

设备	接口	IP 地址	子网掩码	默认网关	DLCI
R1（AR1220）	Serial 1/0/0	11.1.1.1	255.255.255.0	N/A	102
	Serial 1/0/0.1	22.1.1.1	255.255.255.0	N/A	103
R2（AR1220）	Serial 1/0/0	11.1.1.2	255.255.255.0	N/A	201
R3（AR1220）	Serial 1/0/0	22.1.1.3	255.255.255.0	N/A	301

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 地址配置。

在帧中继交换机上配置两条 PVC，R1 和 R2 一条，R1 和 R3 一条。

（1）在帧中继交换机上建立一条 PVC，这条 PVC 在 S 0/0/1 接口上分配 DLCI 为 102，在 S 0/0/2 接口上分配 DLCI 201，二者同属于一条 PVC，如图 11-7 所示。

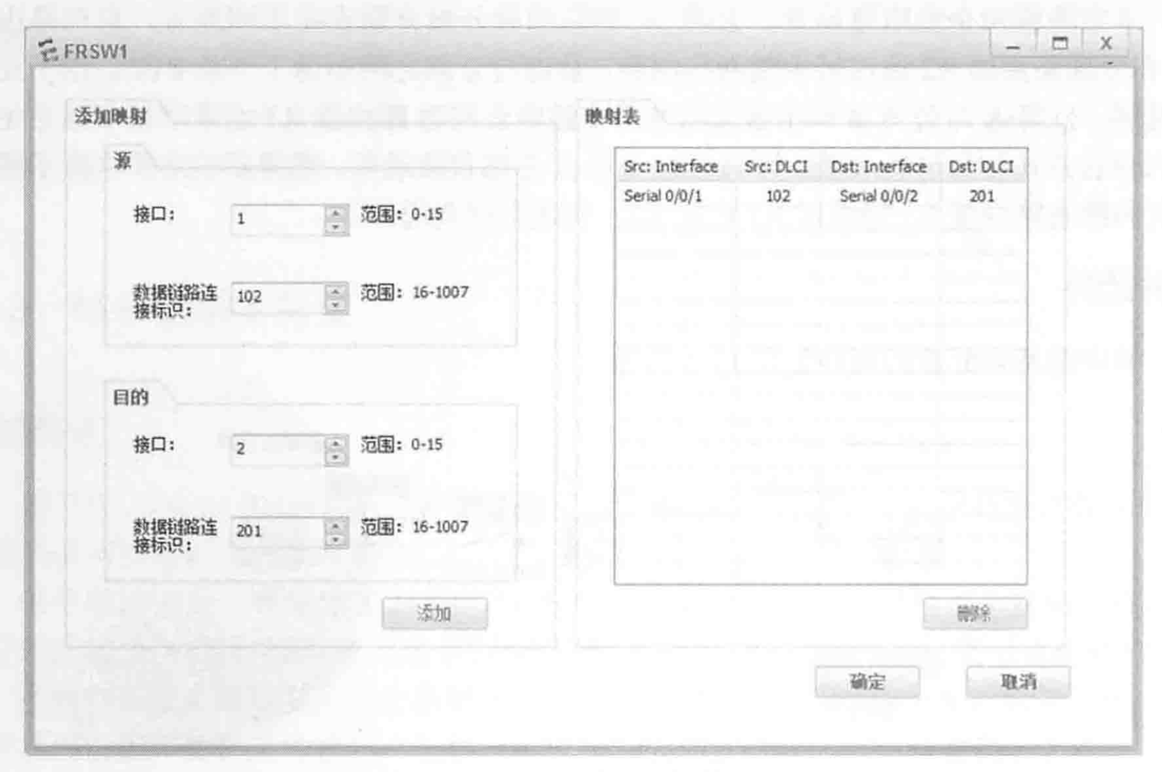


图 11-7 创建 R1 与 R2 间的 PVC

（2）如图 11-8 所示，在帧中继交换机上建立另一条 PVC，这条 PVC 在 S 0/0/1 接口上分配 DLCI 为 103，在 S 0/0/3 接口上分配 DLCI 301，二者同属于另一条 PVC。

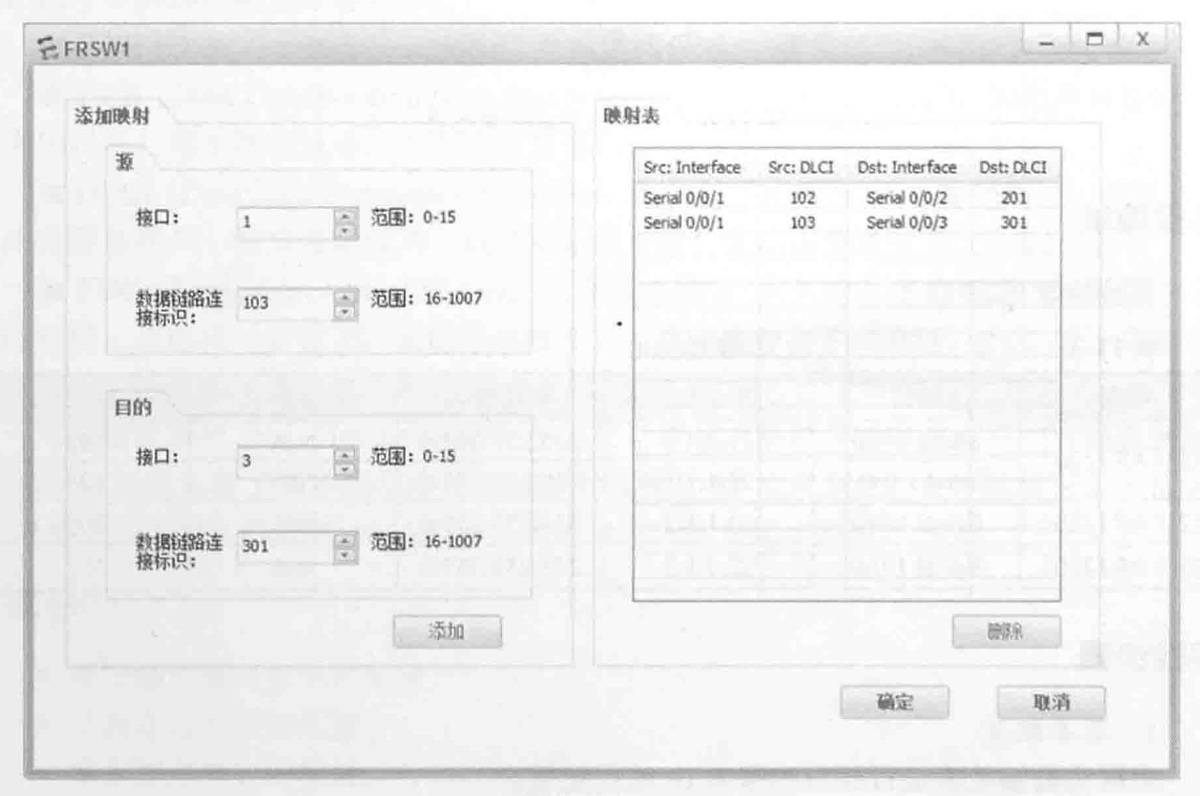


图 11-8 创建 R1 与 R3 间的 PVC

2. 静态与动态映射的配置

帧中继接口在转发数据包时必须查找帧中继地址映射表来确定下一跳的 DLCI。地址映射表中存放对端 IP 地址和下一跳的的 DLCI 的映射关系。只有找到相应的映射表项，才能完成二层帧中继报头的封装，这个机制类似于以太网中的 ARP 机制。该地址映射表可以手动配置（静态），也可以使用 Inverse ARP 协议来自动建立（动态）。

公司 A 总部使用动态映射，在 R1 的 S 1/0/0 接口配置链路层协议为 FR，并使用 **fr inarp** 命令允许帧中继逆向地址解析功能自动生成地址映射表。

```
[R1]interface Serial 1/0/0
[R1-Serial1/0/0]link-protocol fr
Warning: The encapsulation protocol of the link will be changed. Continue? [Y/N]:y
[R1-Serial1/0/0]fr inarp
```

注意，默认情况下，串行接口使用的链路层协议为 PPP 协议，当试图改变接口默认的封装方式的时候，路由器会弹出一个警告，输入“y”即可。此外，帧中继接口的逆向地址解析功能默认是开启的，所以 **fr inarp** 命令可以不配置。

公司 A 分部由于只需要与总部通信即可，使用静态映射，在 R2 的 S 1/0/0 接口下配置链路层协议为 FR，关闭逆向解析功能，使用 **fr map ip** 命令手工配置 R1 的 IP 地址与 DLCI 的静态映射。

```
[R2]interface Serial 1/0/0
[R2-Serial1/0/0]link-protocol fr
Warning: The encapsulation protocol of the link will be changed. Continue? [Y/N]:y
[R2-Serial1/0/0]undo fr inarp
[R2-Serial1/0/0]fr map ip 10.1.1.1 201
```

将 R1 的 IP 地址与 R2 本端 DLCI 201 配置为一条静态地址映射，即 R2 通过下一跳 DLCI 201 来访问 R1。

默认情况下，帧中继不支持广播或组播数据的转发。如果需要在帧中继上运行一些动态路由协议，比如 OSPF 协议，需要在静态映射后面添加 broadcast 参数，从而使 PVC 能够正常发送来自路由协议的广播或组播流量。

```
[R2]interface Serial 1/0/0
[R2-Serial1/0/0]fr map ip 11.1.1.1 201 broadcast
```

配置完成后，在 R1 和 R2 上使用 **display fr pvc-info** 命令查看 PVC 的建立情况。

```
<R1>display fr pvc-info
PVC statistics for interface Serial1/0/0 (DTE, physical UP)
DLCI = 102, USAGE = UNUSED (00000000), Serial1/0/0
create time = 2013/06/28 07:11:45, status = ACTIVE
    InARP = Enable, PVC-GROUP = NONE
    in packets = 0, in bytes = 0
    out packets = 50, out bytes = 1500

DLCI = 103, USAGE = UNUSED (00000000), Serial1/0/0
create time = 2013/06/28 07:11:45, status = ACTIVE
    InARP = Enable, PVC-GROUP = NONE
    in packets = 0, in bytes = 0
    out packets = 45, out bytes = 1350

<R2>display fr pvc-info
PVC statistics for interface Serial1/0/0 (DTE, physical UP)
DLCI = 201, USAGE = LOCAL (00000100), Serial1/0/0
```

```
create time = 2013/06/28 07:11:45, status = ACTIVE
    InARP = Disable, PVC-GROUP = NONE
    in packets = 0, in bytes = 0
    out packets = 50, out bytes = 1500
```

可以观察到，R1 上有两条 PVC，而且都为激活状态。R2 上的 PVC 也为激活状态。

使用 **ping** 命令测试 R1 与 R2 之间的连通性。

```
<R1>ping 11.1.1.2
  PING 11.1.1.2: 56  data bytes, press CTRL_C to break
    Reply from 11.1.1.2: bytes=56 Sequence=1 ttl=255 time=540 ms
    Reply from 11.1.1.2: bytes=56 Sequence=2 ttl=255 time=60 ms
    Reply from 11.1.1.2: bytes=56 Sequence=3 ttl=255 time=70 ms
    Reply from 11.1.1.2: bytes=56 Sequence=4 ttl=255 time=20 ms
    Reply from 11.1.1.2: bytes=56 Sequence=5 ttl=255 time=50 ms
    ......
```

此时 R1 和 R2 已经正常通信。

3. 子接口配置和静态路由

由于业务需要，公司 B 需和公司 A 互相通信。

公司 B 和公司 A 总部之间互连 IP 网段使用 22.1.1.0/24。在 R3 的 S 1/0/0 接口配置链路层协议为 FR，并保持默认开启的逆向地址解析功能。

```
[R3]interface serial 1/0/0
[R3-Serial1/0/0]ip address 22.1.1.3 24
[R3-Serial1/0/0]link-protocol fr
```

配置完成后，在 R3 上使用 **display fr pvc-info** 命令查看 PVC 建立的情况。

```
<R3>display fr pvc-info
PVC statistics for interface Serial1/0/0 (DTE, physical UP)
DLCI = 301, USAGE = UNUSED (00000000), Serial1/0/0
create time = 2013/06/28 08:07:06, status = ACTIVE
    InARP = Enable, PVC-GROUP = NONE
    in packets = 0, in bytes = 0
    out packets = 65, out bytes = 2298
```

可以观察到，此时 R3 的 PVC 已经激活。

为实现与 R3 的互相通信，需要在 R1 上创建子接口 S 1/0/0.1，配置与 R3 同网段的 IP 地址，并手工指定本地 DLCI 配置虚电路。

```
[R1]interface Serial 1/0/0.1
[R1-Serial1/0/0.1]ip address 22.1.1.1 24
[R1-Serial1/0/0.1]fr dlci 103
```

默认情况下，帧中继交换机分配的 DLCI 都关联到用户设备的物理接口上，而子接口关联的 DLCI 需要手动指定。

配置完成后，测试 R1 与 R3 间能否正常通信。

```
<R1>ping 22.1.1.3
  PING 22.1.1.3: 56  data bytes, press CTRL_C to break
    Reply from 22.1.1.3: bytes=56 Sequence=1 ttl=255 time=50 ms
    Reply from 22.1.1.3: bytes=56 Sequence=2 ttl=255 time=40 ms
    Reply from 22.1.1.3: bytes=56 Sequence=3 ttl=255 time=50 ms
    Reply from 22.1.1.3: bytes=56 Sequence=4 ttl=255 time=40 ms
    Reply from 22.1.1.3: bytes=56 Sequence=5 ttl=255 time=30 ms
......
```

可以观察到，R1 和 R3 已经正常通信。测试 R2 与 R3 间能否正常通信。

```
<R3>ping 11.1.1.2
```

```
    PING 11.1.1.2: 56    data bytes, press CTRL_C to break
      Request time out
      Request time out
      Request time out
      Request time out
      Request time out
  ......
```

无法正常通信。这是因为 R2 与 R3 不在同一个网段上，需要有到达对方的路由才能连通。

为了使 R2 和 R3 能够互相通信，在 R3 上配置静态路由，目的地址为 R2，下一跳为 R1 的子接口地址；同样在 R2 上也需要配置静态路由，目的地址为 R3，下一跳为 R1 的 S 1/0/0 接口地址。

```
[R3]ip route-static 11.1.1.2 32 22.1.1.1

[R2]ip route-static 22.1.1.3 32 11.1.1.1
```

由此使 R2 与 R3 之间可以通过 R1 来通信，使用 **ping** 命令检查它们之间的连通性。

```
<R3>ping 11.1.1.2
  PING 11.1.1.2: 56    data bytes, press CTRL_C to break
    Reply from 11.1.1.2: bytes=56 Sequence=1 ttl=254 time=110 ms
    Reply from 11.1.1.2: bytes=56 Sequence=2 ttl=254 time=90 ms
    Reply from 11.1.1.2: bytes=56 Sequence=3 ttl=254 time=110 ms
    Reply from 11.1.1.2: bytes=56 Sequence=4 ttl=254 time=90 ms
    Reply from 11.1.1.2: bytes=56 Sequence=5 ttl=254 time=90 ms
......
```

可以观察到，R2 和 R3 间已经能够正常通信。使用 **tracert** 命令查看它们之间的路径。

```
<R3>tracert 11.1.1.2
 traceroute to    11.1.1.2(10.1.1.2), max hops: 30 ,packet length: 40,press CTRL_C to break
 1 22.1.1.1 50 ms    40 ms    40 ms
 2 11.1.1.2 140 ms    90 ms    70 ms
```

可以观察到，R3 去往 R2 的流量经过了 R1。至此，公司 A 与公司 B 所有的设备间都能正常通信。

思考

帧中继中动态映射的过程是怎样的？它和 ARP 机制的区别在哪里？

11.4　OSPF 在帧中继网络中的配置

原理概述

OSPF 将网络分为 4 种不同的类型，即 Point-to-Point、Broadcast、NBMA 及 Point-to-MultiPoint，不同网络类型下 OSPF 的工作机制不一样。比如，在 Broadcast 网络中，OSPF 能够直接建立邻居邻接关系；在 NBMA 网络中默认必须手工指定邻居等。在实际网络中，可通过配置接口的网络类型来强制改变默认的接口的网络类型。在帧中继的环境中，OSPF 默认的网络类型是 NBMA。

实验目的

- 掌握 OSPF 在帧中继网络中的配置方法
- 理解 Hub-Spoke 组网架构
- 掌握在帧中继网络中排除 OSPF 故障的方法

场景

某公司的网络使用 OSPF 协议，该公司由一个总部和两个分支机构组成。R1 为总部路由器，R2 和 R3 分别是两个分支机构的出口路由器。两分支机构都是通过租用运营商的帧中继虚电路来与总部通信的。但为了节省成本，两个分支机构间没有直接互联的虚电路，即典型的 Hub-Spoke 组网架构，R1 为 Hub 端设备，R2、R3 为 Spoke 端设备。

实验拓扑

配置 OSPF 帧中继网络的拓扑如图 11-9 所示。

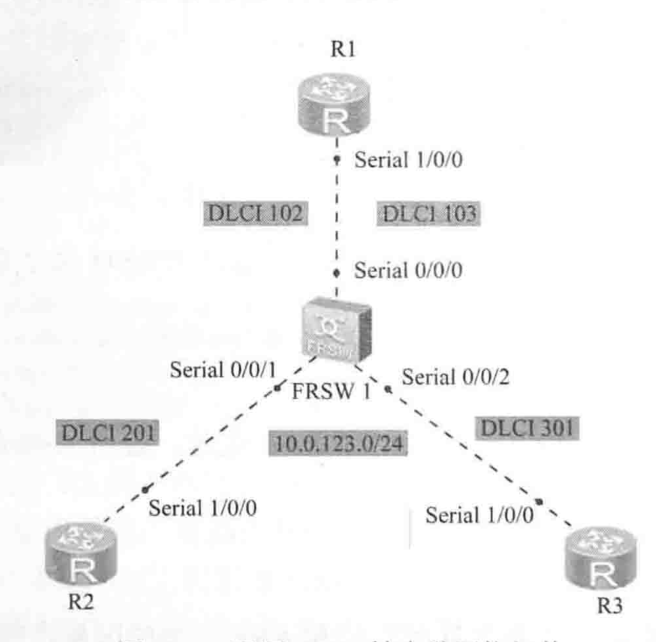


图 11-9 配置 OSPF 帧中继网络拓扑

实验编址

实验编址见表 11-4。

表 11-4 实验编址

设备	接口	IP 地址	子网掩码	默认网关	DLCI
R1（AR2220）	Loopback 0	10.1.1.1	255.255.255.255	N/A	N/A
	Serial 1/0/0	10.0.123.1	255.255.255.0	N/A	102/103
R2（AR2220）	Loopback 0	10.1.2.2	255.255.255.255	N/A	N/A
	Serial 1/0/0	10.0.123.2	255.255.255.0	N/A	201
R3（AR2220）	Loopback 0	10.1.3.3	255.255.255.255	N/A	N/A
	Serial 1/0/0	10.0.123.3	255.255.255.0	N/A	301

实验步骤

1. 基本配置

在公司总部路由器 R1 和两个分部的路由器 R2、R3 上配置帧中继接口，关闭帧中继逆向地址解析功能。

首先根据实验编址表进行相应的基本 IP 地址配置，并配置帧中继静态地址映射。环回接口的掩码为 32 位，用来模拟公司总部和分部的主机。注意将 R1 设置为 DR，调整其 DR 优先级为 100。

```
[R1]interface Serial 1/0/0
[R1-Serial1/0/0]link-protocol fr
[R1-Serial1/0/0]ip address 10.0.123.1 24
[R1-Serial1/0/0]undo fr inarp
[R1-Serial1/0/0]fr map ip 10.0.123.2 102
[R1-Serial1/0/0]fr map ip 10.0.123.3 103
[R1-Serial1/0/0]ospf dr-priority 100
[R1-Serial1/0/0]interface loopback 0
[R1-LoopBack0]ip address 10.1.1.1 32

[R2]interface Serial 1/0/0
[R2-Serial1/0/0]link-protocol fr
[R2-Serial1/0/0]ip address 10.0.123.2 24
[R2-Serial1/0/0]undo fr inarp
[R2-Serial1/0/0]fr map ip 10.0.123.1 201
[R2-Serial1/0/0]interface LoopBack 0
[R2-LoopBack0]ip address 10.1.2.2 32

[R3]interface Serial 1/0/0
[R3-Serial1/0/0]link-protocol fr
[R3-Serial1/0/0]ip address 10.0.123.3 24
[R3-Serial1/0/0]undo fr inarp
[R3-Serial1/0/0]fr map ip 10.0.123.1 301
[R3-Serial1/0/0]interface LoopBack 0
[R3-LoopBack0]ip address 10.1.3.3 32
```

配置完成后，检查帧中继的虚电路状态和映射表。

```
<R1>display fr pvc-info
PVC statistics for interface Serial1/0/0 (DTE, physical UP)
 DLCI = 102, USAGE = LOCAL (00000100), Serial1/0/0
     create time = 2013/05/30 20:35:14, status = ACTIVE
     InARP = Disable,PVC-GROUP = NONE
     in BECN = 0, in FECN = 0
     in packets = 1, in bytes = 30
     out packets = 2, out bytes = 60
DLCI = 103, USAGE = LOCAL (00000100), Serial1/0/0
     create time = 2013/05/30 20:35:14, status = ACTIVE
     InARP = Disable,PVC-GROUP = NONE
     in BECN = 0, in FECN = 0
     in packets = 5, in bytes = 440
out packets = 7, out bytes = 500
```

可以观察到，PVC 处于 ACTIVE 状态表示正常。

```
<R1>display fr map-info
Map Statistics for interface Serial1/0/0 (DTE)
```

```
    DLCI = 102, IP 10.0.123.2, Serial1/0/0
      create time = 2013/05/30 20:35:41, status = ACTIVE
encapsulation = ietf, vlink = 1
    DLCI = 103, IP 10.0.123.3, Serial1/0/0
      create time = 2013/05/30 20:35:52, status = ACTIVE
encapsulation = ietf, vlink = 2
```

检查 R1 与 R2，R1 与 R3 间的网络连通性。

```
<R1>ping 10.0.123.2
  PING 10.0.123.2: 56  data bytes, press CTRL_C to break
    Reply from 10.0.123.2: bytes=56 Sequence=1 ttl=255 time=60 ms
    Reply from 10.0.123.2: bytes=56 Sequence=2 ttl=255 time=20 ms
    Reply from 10.0.123.2: bytes=56 Sequence=3 ttl=255 time=40 ms
    Reply from 10.0.123.2: bytes=56 Sequence=4 ttl=255 time=30 ms
    Reply from 10.0.123.2: bytes=56 Sequence=5 ttl=255 time=30 ms
  --- 10.0.123.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 20/36/60 ms

<R1>ping 10.0.123.3
  PING 10.0.123.3: 56  data bytes, press CTRL_C to break
    Reply from 10.0.123.3: bytes=56 Sequence=1 ttl=255 time=30 ms
    Reply from 10.0.123.3: bytes=56 Sequence=2 ttl=255 time=30 ms
    Reply from 10.0.123.3: bytes=56 Sequence=3 ttl=255 time=50 ms
    Reply from 10.0.123.3: bytes=56 Sequence=4 ttl=255 time=40 ms
    Reply from 10.0.123.3: bytes=56 Sequence=5 ttl=255 time=40 ms
  --- 10.0.123.3 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 30/38/50 ms
```

此时通信正常。

2. 在帧中继上搭建 OSPF 网络

在 R1、R2 和 R3 上配置 OSPF 协议。由于网络拓扑简单，采用 OSPF 的单区域配置即可，指定它们各自的环回接口地址作为 Router-ID，所有网段都属于区域 0。

```
[R1]ospf 1 router-id 10.1.1.1
[R1-ospf-1]area 0
[R1-ospf-1-area-0.0.0.0]network 10.0.123.1 0.0.0.255
[R1-ospf-1-area-0.0.0.0]network 10.1.1.1 0.0.0.0

[R2]ospf 1 router-id 10.1.2.2
[R2-ospf-1]area 0
[R2-ospf-1-area-0.0.0.0]network 10.0.123.2 0.0.0.255
[R2-ospf-1-area-0.0.0.0]network 10.1.2.2 0.0.0.0

[R3]ospf 1 router-id 10.1.3.3
[R3-ospf-1]area 0
[R3-ospf-1-area-0.0.0.0]network 10.0.123.3 0.0.0.255
```

```
[R3-ospf-1-area-0.0.0.0]network 10.1.3.3 0.0.0.0
```

配置完成后，查看 OSPF 的邻居建立情况。

```
<R1>display ospf peer
          OSPF Process 1 with Router-ID 10.1.1.1
```

发现无法正常建立邻居，这是明显的网络故障，现在网络管理员需要立刻进行分析排除故障。排障的时候需要注意遵循从底层逐步往上层排查的顺序，即先检查物理层线缆是否正常，然后检查二层链路的连通性，再检查三层路由协议的运行情况，最后检查高层相关应用是否正常。

物理层检查这里省略，首先测试直连线路的连通性。

```
[R1]ping 10.0.123.2
  PING 10.0.123.2: 56   data bytes, press CTRL_C to break
    Reply from 10.0.123.2: bytes=56 Sequence=1 ttl=255 time=100 ms
    Reply from 10.0.123.2: bytes=56 Sequence=2 ttl=255 time=10 ms
    Reply from 10.0.123.2: bytes=56 Sequence=3 ttl=255 time=10 ms
    Reply from 10.0.123.2: bytes=56 Sequence=4 ttl=255 time=10 ms
    Reply from 10.0.123.2: bytes=56 Sequence=5 ttl=255 time=10 ms
  --- 10.0.123.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 10/28/100 ms
```

直连链路连通性没有问题。再查看三层路由协议，即相应接口否被通告到 OSPF 进程中。

```
[R1]display ospf interface
          OSPF Process 1 with Router-ID 10.1.1.1
                  Interfaces
 Area: 0.0.0.0           (MPLS TE not enabled)
 IP Address      Type      State    Cost    Pri      DR          BDR
 10.0.123.1      NBMA       DR      48      100    10.0.123.1   0.0.0.0
 10.1.1.1        P2P       P-2-P    0       1       0.0.0.0     0.0.0.0
```

观察到所有接口已经被通告进入 OSPF 进程。

此时可以对 R1 的 S 1/0/0 接口进行抓包分析，查看协议的运行情况，如图 11-10 所示。

```
 1 0.000000     Q.933  STATUS ENQUIRY[Malformed P
 2 0.000000     Q.933  STATUS
 3 10.000000    Q.933  STATUS ENQUIRY[Malformed P
 4 10.000000    Q.933  STATUS
 5 20.000000    Q.933  STATUS ENQUIRY[Malformed P
 6 20.000000    Q.933  STATUS
 7 30.000000    Q.933  STATUS ENQUIRY[Malformed P
 8 30.000000    Q.933  STATUS
 9 40.000000    Q.933  STATUS ENQUIRY[Malformed P
10 40.000000    Q.933  STATUS
11 50.000000    Q.933  STATUS ENQUIRY[Malformed P
12 50.000000    Q.933  STATUS
13 60.000000    Q.933  STATUS ENQUIRY[Malformed P
14 60.000000    Q.933  STATUS
```

图 11-10　抓包观察

发现 R1 始终没有向外发送 OSPF 数据包。这是由于 OSPF 在帧中继上默认的网络类型为 NBMA，即非广播多路访问。这种网络类型的特点是不支持广播和组播的数据包，

而 OSPF 协议默认是采用组播方式发送报文，所以设备的 OSPF 报文无法在帧中继链路上进行发送，导致没有成功建立邻居关系。

这时可以采用 **peer** 命令手工指定 OSPF 邻居，采用单播方式发送报文。

```
[R1]ospf 1
[R1-ospf-1]peer 10.0.123.2
[R1-ospf-1]peer 10.0.123.3

[R2]ospf 1
[R2-ospf-1]peer 10.0.123.1

[R3]ospf 1
[R3-ospf-1]peer 10.0.123.1
```

配置完成后，再次检查 OSPF 的邻居关系状态。

```
<R1>display ospf peer brief
        OSPF Process 1 with Router-ID 10.0.1.1
                Peer Statistic Information
-----------------------------------------------------------------------------
 Area Id          Interface                        Neighbor id        State
0.0.0.0          Serial1/0/0                       10.1.2.2           Full
0.0.0.0          Serial1/0/0                       10.1.3.3           Full
-----------------------------------------------------------------------------
```

可以观察到，这时 R1 与 R2、R3 都建立了完全的邻接关系。再查看 R1、R2、R3 的路由表。

```
<R1>display ip routing protocol ospf
Route Flags: R - relay, D - download to fib
-----------------------------------------------------------------------------
Public routing table : OSPF
          Destinations : 2          Routes : 2
OSPF routing table status : <Active>
          Destinations : 2          Routes : 2
Destination/Mask    Proto   Pre   Cost    Flags   NextHop       Interface
 10.1.2.2/32        OSPF    10    48      D       10.0.123.2    Serial1/0/0
 10.1.3.3/32        OSPF    10    48      D       10.0.123.3    Serial1/0/0
OSPF routing table status : <Inactive>
          Destinations : 0          Routes : 0

<R2>display ip routing protocol ospf
Route Flags: R - relay, D - download to fib
-----------------------------------------------------------------------------
Public routing table : OSPF
          Destinations : 2          Routes : 2
OSPF routing table status : <Active>
          Destinations : 2          Routes : 2
Destination/Mask    Proto   Pre   Cost    Flags   NextHop       Interface
10.1.1.1/32         OSPF    10    48      D       10.0.123.1    Serial1/0/0
10.1.3.3/32         OSPF    10    48      D       10.0.123.3    Serial1/0/0
OSPF routing table status : <Inactive>
          Destinations : 0          Routes : 0
```

```
<R3>display ip routing-table protocol ospf
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Public routing table : OSPF
         Destinations : 2          Routes : 2
OSPF routing table status : <Active>
         Destinations : 2          Routes : 2
Destination/Mask    Proto    Pre    Cost    Flags    NextHop        Interface
 10.1.1.1/32        OSPF     10     48      D        10.0.123.1     Serial1/0/0
 10.1.2.2/32        OSPF     10     48      D        10.0.123.2     Serial1/0/0
OSPF routing table status : <Inactive>
         Destinations : 0          Routes : 0
```

可以观察到此时的 R1、R2、R3 路由表中都互相接收到了各自环回口所在网段的路由条目。测试环回口之间的连通性。

```
<R1>ping -a 10.1.1.1 10.1.2.2
  PING 10.1.2.2: 56   data bytes, press CTRL_C to break
    Reply from 10.1.2.2: bytes=56 Sequence=1 ttl=255 time=10 ms
    Reply from 10.1.2.2: bytes=56 Sequence=2 ttl=255 time=10 ms
    Reply from 10.1.2.2: bytes=56 Sequence=3 ttl=255 time=10 ms
    Reply from 10.1.2.2: bytes=56 Sequence=4 ttl=255 time=20 ms
    Reply from 10.1.2.2: bytes=56 Sequence=5 ttl=255 time=10 ms
  --- 10.1.2.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 10/12/20 ms

<R1>ping -a 10.1.1.1 10.1.3.3
  PING 10.1.3.3: 56   data bytes, press CTRL_C to break
    Reply from 10.1.3.3: bytes=56 Sequence=1 ttl=255 time=20 ms
    Reply from 10.1.3.3: bytes=56 Sequence=2 ttl=255 time=10 ms
    Reply from 10.1.3.3: bytes=56 Sequence=3 ttl=255 time=10 ms
    Reply from 10.1.3.3: bytes=56 Sequence=4 ttl=255 time=10 ms
    Reply from 10.1.3.3: bytes=56 Sequence=5 ttl=255 time=10 ms
  --- 10.1.3.3 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 10/12/20 ms
```

此时 R1 与 R2，R1 与 R3 之间的环回口通信正常。测试 R2 与 R3 环回口之间的通信情况。

```
<R2>ping -a 10.1.2.2 10.1.3.3
  PING 10.1.3.3: 56   data bytes, press CTRL_C to break
    Request time out
    Request time out
    Request time out
    Request time out
    Request time out
……
```

发现 R2 无法连通 R3 的环回口。此时网络管理员需要再次进行排障。物理链路和二层链路的连通性测试这里省略。首先查看 R2 上的 OSPF 路由条目，观察到去往 10.1.3.3 的网段下一跳地址是 10.0.123.3。

```
<R2>display ip routing protocol ospf
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Public routing table : OSPF
         Destinations : 2          Routes : 2
OSPF routing table status : <Active>
         Destinations : 2          Routes : 2
Destination/Mask    Proto   Pre   Cost    Flags   NextHop       Interface
10.1.1.1/32         OSPF    10    48      D       10.0.123.1    Serial1/0/0
10.1.3.3/32         OSPF    10    48      D       10.0.123.3    Serial1/0/0
OSPF routing table status : <Inactive>
         Destinations : 0          Routes : 0
```

然后在 R2 上查看帧中继映射关系。

```
<R2>display fr map-info
Map Statistics for interface Serial1/0/0 (DTE)
  DLCI = 201, IP 10.0.123.1, Serial1/0/0
    create time = 2013/06/23 20:06:07, status = ACTIVE
encapsulation = ietf, vlink = 1
```

可以观察到，此时没有关于 10.0.123.3 的映射，如果 R2 要发送数据包至下一跳 10.0.123.3，但无法知晓该从哪条 PVC 上进行发送和封装，可以使用 PVC 复用技术解决此问题。这时需在 R1 的 S 1/0/0 接口下添加一条帧中继静态映射，通过 R1 与 R2 的 PVC 去往 10.0.123.3。

```
[R2]interface Serial 1/0/0
[R2-Serial1/0/0]fr map ip 10.0.123.3 201
```

同样需要在 R3 上也添加关于 10.0.123.2 的相关映射。

```
[R3]interface Serial 1/0/0
[R3-Serial1/0/0]fr map ip 10.0.123.2 301
```

配置完成后，再在 R2 上查看帧中继映射关系。

```
<R2>display fr map-info
Map Statistics for interface Serial1/0/0 (DTE)
  DLCI = 201, IP 10.0.123.1, Serial1/0/0
    create time = 2013/06/23 20:06:07, status = ACTIVE
encapsulation = ietf, vlink = 1
DLCI = 201, IP 10.0.123.3, Serial1/0/0
    create time = 2013/06/23 20:06:07, status = ACTIVE
encapsulation = ietf, vlink = 2
```

此时可以观察到已经添加上了相应映射。

再次测试 R2 到 R3 环回口的连通性。

```
[R2]ping -a 10.1.2.2 10.1.3.3
  PING 10.1.3.3: 56  data bytes, press CTRL_C to break
    Reply from 10.1.3.3: bytes=56 Sequence=1 ttl=254 time=50 ms
    Reply from 10.1.3.3: bytes=56 Sequence=2 ttl=254 time=20 ms
    Reply from 10.1.3.3: bytes=56 Sequence=3 ttl=254 time=30 ms
    Reply from 10.1.3.3: bytes=56 Sequence=4 ttl=254 time=20 ms
```

```
    Reply from 10.1.3.3: bytes=56 Sequence=5 ttl=254 time=30 ms
  --- 10.1.3.3 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 20/30/50 ms
```

此时通信正常，所有问题解决。

思考

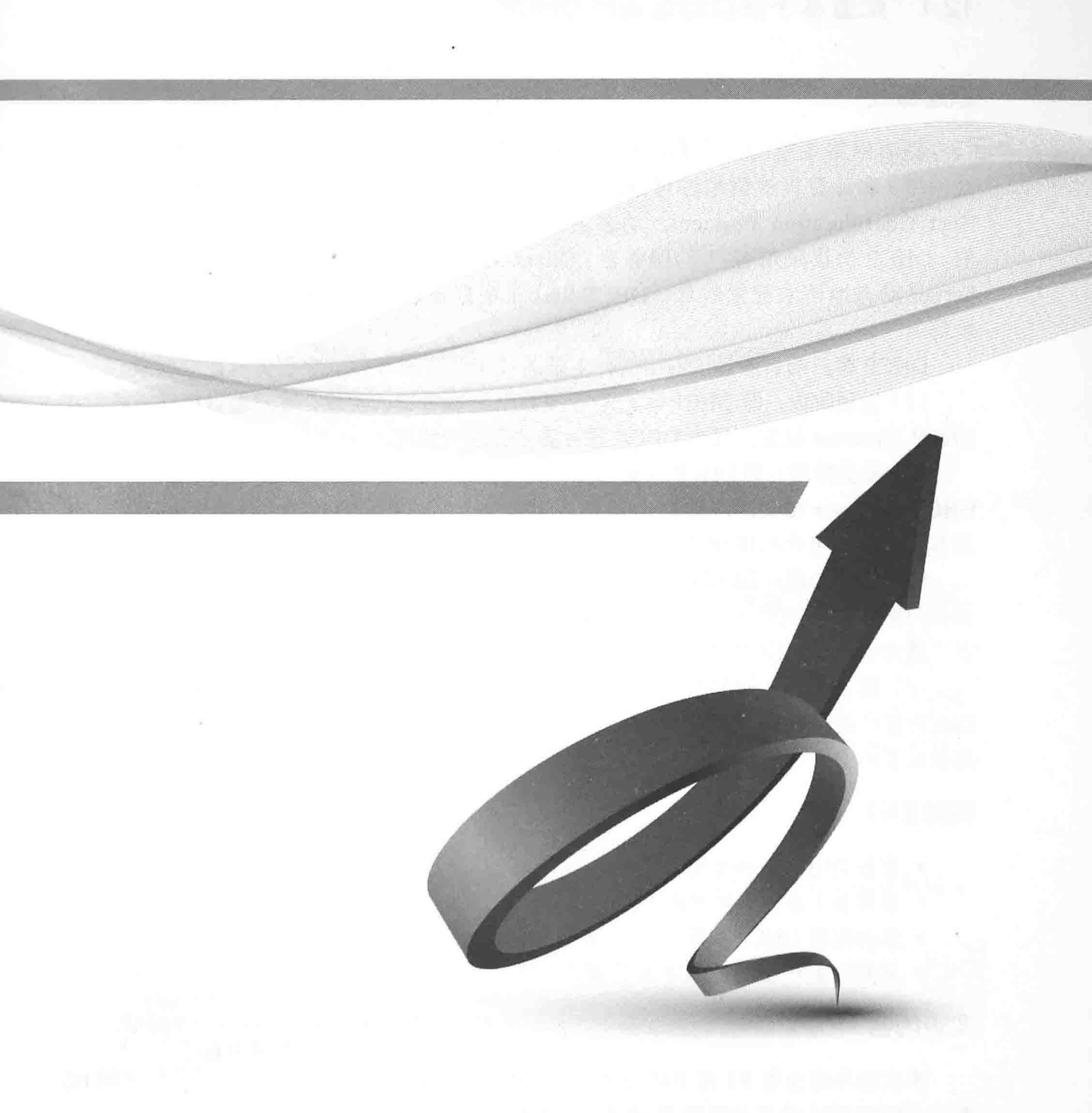
在第 1 步的基本配置中将 R1 的 DR 优先级设置成了 100，为什么要这么做？

第12章 DHCP

12.1 配置基于接口地址池的DHCP

12.2 配置基于全局地址池的DHCP

12.3 配置DHCP中继



12.1 配置基于接口地址池的 DHCP

原理概述

随着网络规模的扩大和网络复杂程度的提高，计算机位置变化（如便携机或无线网络）和计算机数量超过可分配的 IP 地址的情况将会经常出现。DHCP（Dynamic Host Configuration Protocol，动态主机配置协议）就是为满足这些需求而发展起来的。DHCP 协议采用客户端/服务器（Client/Server）方式工作，DHCP Client 向 DHCP Server 动态地请求配置信息，DHCP Server 根据策略返回相应的配置信息（如 IP 地址等）。

DHCP 客户端首次登录网络时，主要通过 4 个阶段与 DHCP 服务器建立联系。

（1）发现阶段：即 DHCP 客户端寻找 DHCP 服务器的阶段。客户端以广播方式发送 DHCP_Discover 报文，只有 DHCP 服务器才会进行响应。

（2）提供阶段：即 DHCP 服务器提供 IP 地址的阶段。DHCP 服务器接收到客户端的 DHCP_Discover 报文后，从 IP 地址池中挑选一个尚未分配的 IP 地址分配给客户端，向该客户端发送包含出租 IP 地址和其他设置的 DHCP_Offer 报文。

（3）选择阶段：即 DHCP 客户端选择 IP 地址的阶段。如果有多台 DHCP 服务器向该客户端发来 DHCP_Offer 报文，客户端只接受第一个收到的 DHCP_Offer 报文，然后以广播方式向各 DHCP 服务器回应 DHCP_Request 报文。

（4）确认阶段：即 DHCP 服务器确认所提供 IP 地址的阶段。当 DHCP 服务器收到 DHCP 客户端回答的 DHCP_Request 报文后，便向客户端发送包含它所提供的 IP 地址和其他设置的 DHCP_ACK 确认报文。

实验目的

- 掌握 DHCP Server 配置方法
- 掌握基于接口地址池的 DHCP Server 配置方法
- 掌握配置 DHCP 租期/不参与自动分配地址/DNS 服务器地址方法
- 掌握配置和检测 DHCP 客户端的方法

实验内容

本实验将路由器 R1 模拟成为公司的 DHCP Server，该公司市场部和财务部下的 PC 通过 DHCP 的方式自动配置 IP 地址。网络管理员配置客户端 PC 通过接口地址池的方式自动获取 IP 地址。

实验拓扑

配置基于接口地址池的 DHCP 拓扑如图 12-1 所示。

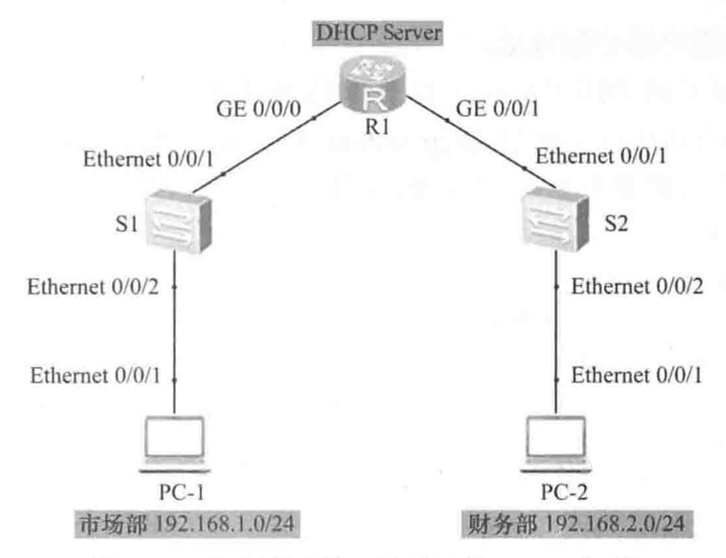


图 12-1　配置基于接口地址池的 DHCP 拓扑

实验编址

实验编址见表 12-1。

表 12-1　　实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR2220）	GE 0/0/0	192.168.1.254	255.255.255.0	N/A
	GE 0/0/1	192.168.2.254	255.255.255.0	N/A
PC-1	Ethernet0/0/1	DHCP 获取	DHCP 获取	DHCP 获取
PC-2	Ethernet0/0/1	DHCP 获取	DHCP 获取	DHCP 获取

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 地址配置，由于 PC 是通过 DHCP 自动获取地址，暂时无法测试连通性。交换机为二层设备，无需配置 IP 地址。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]ip address 192.168.1.254 24
[R1-GigabitEthernet0/0/0]interface GigabitEthernet 0/0/1
[R1-GigabitEthernet0/0/1]ip address 192.168.2.254 24
```

2. 基于接口配置 DHCP Server 功能

在 R1 上开启 DHCP 功能。

```
[R1]dhcp enable
```

在 R1 的 GE 0/0/0 和 GE 0/0/1 接口上配置 **dhcp select interface** 命令，开启接口的 DHCP 服务功能，指定从接口地址池分配地址。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]dhcp select interface
[R1-GigabitEthernet0/0/0]interface GigabitEthernet 0/0/1
[R1-GigabitEthernet0/0/1]dhcp select interface
```

接口地址池可动态分配 IP 地址，范围就是接口的 IP 地址所在网段，且只在此接口下有效。当 DHCP 服务器接收到 DHCP 客户端的请求报文后，DHCP 服务器将会使用该

接口的地址网段给客户端分配地址。

3. 配置基于接口的 DHCP Server 租期/DNS 服务器地址

在 R1 的 GE 0/0/0 接口上使用 **dhcp server lease** 命令配置 DHCP 服务器接口地址池中 IP 地址的租用有效期限为 2 天，GE 0/0/1 接口不修改，使用默认值 1 天，超过租期后该地址将会重新分配。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]dhcp server lease day 2
```

在 GE 0/0/0 接口上使用 **dhcp server excluded-ip-address** 命令配置接口地址池中不参与自动分配的 IP 地址范围为 192.168.1.1 到 192.168.1.10。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]dhcp server excluded-ip-address 192.168.1.1 192.168.1.10
```

有些地址需要分配给其他服务，如 DNS 服务器或 HTTP 服务器等需要手工静态配置的 IP 地址，就不能再动态分配给客户端使用，可以执行该命令配置地址池中不参与自动分配的 IP 地址（默认该地址池所有地址参与自动分配，此命令作为可选命令）。

当 DHCP 服务器收到客户端的 DHCP 请求时，DPCP 服务器将会选择地址池中空闲的 IP 地址分配给客户端。GE 0/0/0 接口地址池中 192.168.1.1～192.168.1.10 不参与分配，而 GE 0/0/1 接口没有配置该命令，因此可以分配的 IP 地址范围是 192.168.2.1～192.168.2.253（不包括本接口地址）。

在 GE 0/0/1 接口上使用 **dhcp server dns-list** 命令指定接口地址池下的 DNS 服务器，为 PC-2 自动分配 DNS 服务器地址为 8.8.8.8。

```
[R1-GigabitEthernet0/0/1]dhcp server dns-list 8.8.8.8
```

4. 配置 DHCP Client

打开 PC-1 的“基础配置”选项卡，在“IPv4 配置”栏中选择“DHCP”，然后单击对话框右下角的“应用”按钮，如图 12-2 所示。

图 12-2 PC-1 配置界面

单击 PC-1 的“命令行”选项卡，在其中输入 **“ipconfig”** 命令查看接口的 IP 地址，如图 12-3 所示。

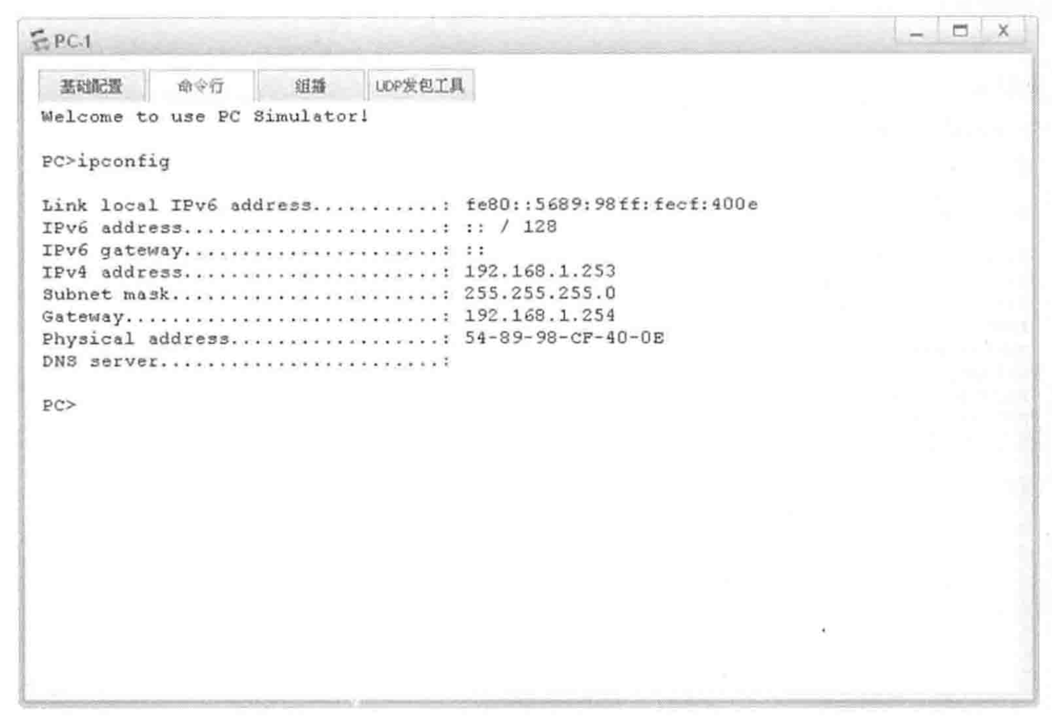
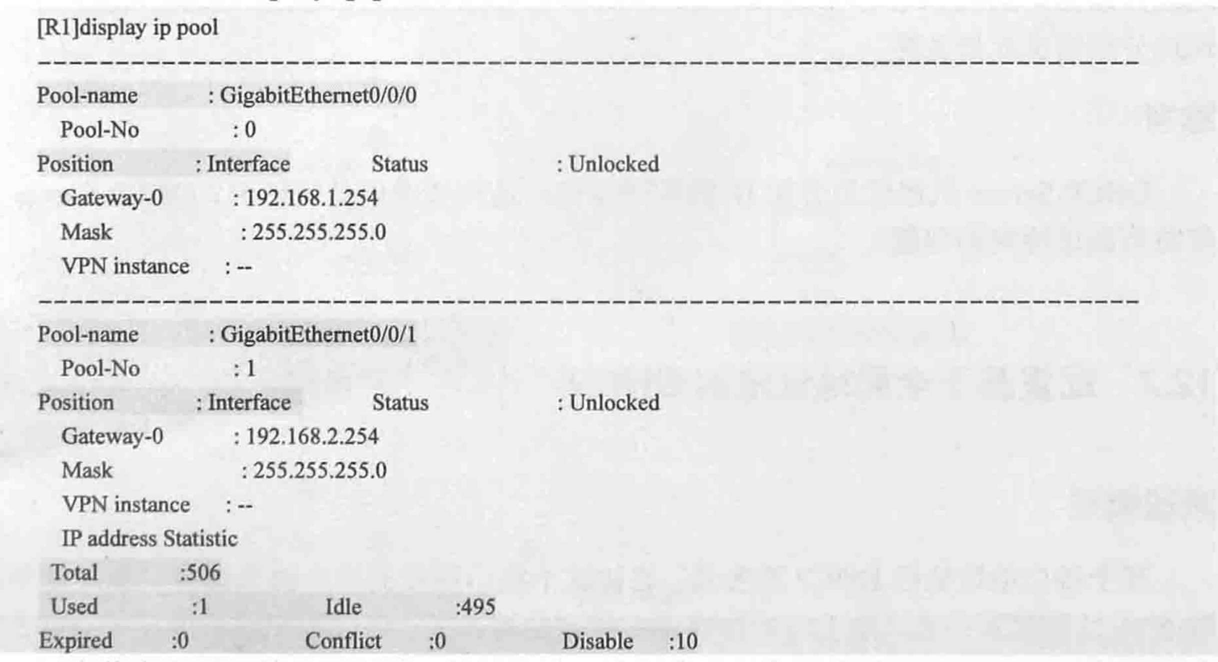


图 12-3 查看 PC-1 的 IP 地址

通过观察发现 PC-1 已经通过 DHCP Server 获取到一个 IPv4 地址 192.168.1.253，网关地址为路由器的接口地址 192.168.1.254。

在 R1 上使用 **display ip pool** 命令查看 DHCP 地址池中的地址分配情况。



```
[R1]display ip pool
-----------------------------------------------------------------------------
Pool-name         : GigabitEthernet0/0/0
  Pool-No         : 0
Position        : Interface          Status              : Unlocked
  Gateway-0       : 192.168.1.254
  Mask            : 255.255.255.0
  VPN instance    : --
-----------------------------------------------------------------------------
Pool-name         : GigabitEthernet0/0/1
  Pool-No         : 1
Position        : Interface          Status              : Unlocked
  Gateway-0       : 192.168.2.254
  Mask            : 255.255.255.0
  VPN instance    : --
  IP address Statistic
  Total         :506
  Used          :1          Idle          :495
Expired        :0          Conflict      :0             Disable    :10
```

以上信息显示目前为基于接口的地址池，由于有两个接口启用 DHCP 功能，所以地址池也有两个，Pool-name 分别为 GE 0/0/0、GE 0/0/1。在 DHCP Server 地址池中，网关为 192.168.1.254，掩码为 255.255.255.0，IP 地址池总共可以分配 506 个地址（除了路由器接口地址），已经使用了一个，空闲地址为 495 个，其中地址池中有 10 个地址是不参与分配的。

配置 PC-2 时参考配置 PC-1 的方法，选择通过 DHCP 配置地址。

单击 PC-2 中的“命令行”选项卡，在其中输入**“ipconfig”**命令查看接口的 IP 地址，如图 12-4 所示。



图 12-4 查看 PC-2 的 IP 地址

通过观察发现 PC-2 已通过 DHCP Server 获取到一个 IPv4 地址 192.168.2.253，网关地址为路由器的接口地址 192.168.2.254，DNS 服务器地址为 8.8.8.8。DHCP 地址池中的地址分配情况此处省略。

思考

DHCP Server 从地址池分配 IP 的顺序如何，是按顺序还是随机的？DHCP Server 如何防范地址冲突的问题？

12.2 配置基于全局地址池的 DHCP

原理概述

基于接口地址池的 DHCP 服务器，连接这个接口网段的用户都从该接口地址池中获取 IP 地址等配置信息，由于地址池绑定在特定的接口上，可以限制用户的使用条件，因此在保障了安全性的同时也存在一定局限性。当用户从不同接口接入 DHCP 服务器且需要从同一个地址池里获取 IP 地址时，就需要配置基于全局地址池的 DHCP。

配置基于全局地址池的 DHCP 服务器，从所有接口上连接的用户都可以选择该地址池中的地址，也就是说全局地址池是一个公共地址池。在 DHCP 服务器上创建地址池并

配置相关属性（包括地址范围、地址租期、不参与自动分配的 IP 地址等），再配置接口工作在全局地址池模式。路由器支持工作在全局地址池模式的接口有三层接口及其子接口、三层 Ethernet 接口及其子接口、三层 Eth-Trunk 接口及其子接口和 VLANIF 接口。

实验目的

- 掌握 DHCP Server 配置方法
- 掌握基于全局地址池的 DHCP Server 配置方法
- 掌握配置 DHCP 租期/网关地址/不参与自动分配地址方法
- 掌握配置和检测 DHCP 客户端的方法

场景

本实验将路由器 R1 模拟成公司 DHCP Server，配置全局地址池，该公司市场部和财务部下的 PC 通过 DHCP 的方式自动配置 IP 地址。

实验拓扑

配置基于全局地址池的 DHCP 拓扑如图 12-5 所示。

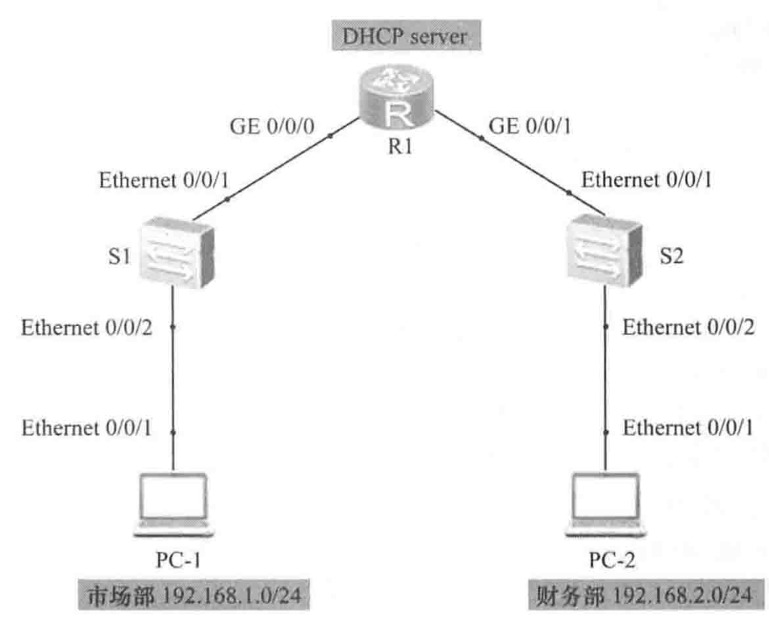


图 12-5 配置基于全局地址池的 DHCP 拓扑

实验编址

实验编址见表 12-2。

表 12-2 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR2220）	GE 0/0/0	192.168.1.254	255.255.255.0	N/A
	GE 0/0/1	192.168.2.254	255.255.255.0	N/A
PC-1	Ethernet 0/0/1	DHCP 获取	DHCP 获取	DHCP 获取
PC-2	Ethernet 0/0/1	DHCP 获取	DHCP 获取	DHCP 获取

实验步骤

1．基本配置

根据实验编址表进行相应的基本IP地址配置，由于PC是通过DHCP自动获取地址，暂时无法测试连通性。交换机为二层设备，无需配置IP地址。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]ip address 192.168.1.254 24
[R1-GigabitEthernet0/0/0]interface GigabitEthernet 0/0/1
[R1-GigabitEthernet0/0/1]ip address 192.168.2.254 24
```

2．配置基于全局地址池的DHCP Server

在R1上开启DHCP功能。

```
[R1]dhcp enable
```

使用**ip pool**命令创建一个全局地址池，地址池名称为huawei1。默认情况下，设备上没有创建任何全局地址池。

```
[R1]ip pool huawei1
```

使用**network**命令配置全局地址池huawei1可动态分配的网段范围为192.168.1.0，如果不指定掩码，则默认使用自然掩码，即24位掩码。该网段必须与路由器接口GE 0/0/0的IP地址为同一网段。

```
[R1-ip-pool-huawei1]network 192.168.1.0
```

使用**lease day**命令配置DHCP全局地址池下的地址租期。默认情况下，IP地址租期为1天，对于不同的地址池，DHCP服务器可以指定不同的地址租用期限，但是同一地址池中的地址具有相同的租期。

```
[R1-ip-pool-huawei1]lease day 2
```

配置DHCP客户端的出口网关地址。

```
[R1-ip-pool-huawei]gateway-list 192.168.1.254
```

配置地址池中192.168.1.250到192.168.1.253这些地址不参与自动分配。

```
[R1-ip-pool-huawei1]excluded-ip-address 192.168.1.250 192.168.1.253
```

由于地址192.168.1.250到192.168.1.253不参与自动分配，而网关地址也不参与自动分配，因此DHCP服务器将会从地址池中由192.168.1.249开始往前分配。

配置DNS服务器地址。

```
[R1-ip-pool-huawei1]dns-list 8.8.8.8
```

开启接口的DHCP功能。使用该命令配置设备指定接口采用全局地址池为客户端分配IP地址。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]dhcp select global
```

路由器需要为两个不同部门分配IP地址，即需要两个全局地址池。为财务部配置的全局地址池名称为huawei2，IP网段为192.168.2.0，网关地址为192.168.2.254，DNS服务器地址为8.8.8.8。配置完成后在GE 0/0/1接口下启用全局地址池的DHCP服务器模式。

```
[R1]ip pool huawei2
[R1-ip-pool-huawei2]network 192.168.2.0
[R1-ip-pool-huawei2]lease day 2
[R1-ip-pool-huawei2]gateway-list 192.168.2.254
[R1-ip-pool-huawei2]dns-list 8.8.8.8
```

```
[R1-ip-pool-huawei2]interface GigabitEthernet 0/0/1
[R1-GigabitEthernet0/0/1]dhcp select global
```

配置完成后，查看 IP 地址池信息。

```
[R1]display ip pool
-----------------------------------------------------------------------------
Pool-name        : huawei1
  Pool-No          : 0
  Position         : Local           Status            : Unlocked
  Gateway-0        : 192.168.1.254
  Mask             : 255.255.255.0
  VPN instance     : --
-----------------------------------------------------------------------------
 Pool-name        : huawei2
  Pool-No          : 1
  Position         : Local           Status            : Unlocked
  Gateway-0        : 192.168.2.254
  Mask             : 255.255.255.0
  VPN instance     : --
  IP address Statistic
Total        :506
Used         :0          Idle         :502
Expired      :0          Conflict     :0          Disable    :4
```

以上信息显示有两个地址池，其中一个地址池为 huawei1，另一个地址池为 huawei2，地址池的总数为 506 个，使用了 0 个，空闲 502 个，有 4 个地址不参与分配。

3. 配置 DHCP Client

打开 PC-1 的“基础配置”选项卡，在“IPV4 配置”栏中选择“DHCP”，然后单击对话框右下角的“应用”按钮，如图 12-6 所示。



图 12-6 PC-1 配置界面

单击 PC-1 的“命令行”选项卡，在其中输入**“ipconfig”**命令查看接口的 IP 地址，如图 12-7 所示。

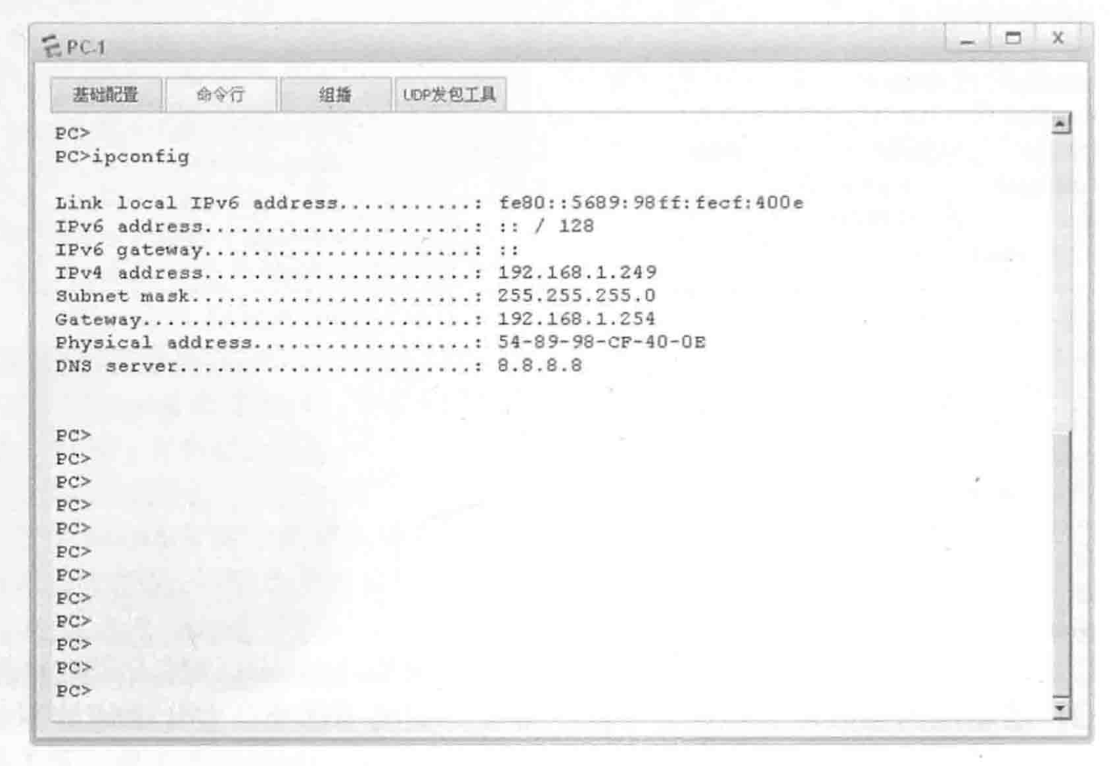


图 12-7 查看 PC-1 的 IP 地址

通过观察发现 PC-1 已经通过 DHCP Server 获取到一个 IPv4 地址 192.168.1.249，网关地址为 192.168.1.254，DNS 服务器地址为 8.8.8.8。

验证路由器与 PC 之间的连通性。

```
[R1]ping 192.168.1.249
  PING 192.168.1.249: 56    data bytes, press CTRL_C to break
    Reply from 192.168.1.249: bytes=56 Sequence=1 ttl=128 time=500 ms
    Reply from 192.168.1.249: bytes=56 Sequence=2 ttl=128 time=180 ms
    Reply from 192.168.1.249: bytes=56 Sequence=3 ttl=128 time=130 ms
    Reply from 192.168.1.249: bytes=56 Sequence=4 ttl=128 time=90 ms
    Reply from 192.168.1.249: bytes=56 Sequence=5 ttl=128 time=110 ms
  --- 192.168.1.249 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 90/202/500 ms
```

可以正常通信。

打开 PC-2 的“基础配置”选项卡，在“IPv4 配置”栏中选择“DHCP”，然后单击对话框右下角的“应用”按钮，如图 12-8 所示。

图 12-8 PC-2 配置界面

单击 PC-2 的“命令行”选项卡，输入**“ipconfig”**命令查看接口的 IP 地址，如图 12-9 所示。

```
Welcome to use PC Simulator!

PC>ipconfig

Link local IPv6 address...........: fe80::5689:98ff:fecf:c54e
IPv6 address......................: :: / 128
IPv6 gateway......................: ::
IPv4 address......................: 192.168.2.253
Subnet mask.......................: 255.255.255.0
Gateway...........................: 192.168.2.254
Physical address..................: 54-89-98-CF-C5-4E
DNS server........................: 8.8.8.8


PC>
```

图 12-9 查看 PC-2 的 IP 地址

通过观察发现 PC-2 已经通过 DHCP Server 获取到一个 IPv4 地址 192.168.2.253，网关地址为 192.168.2.254，DNS 服务器地址为 8.8.8.8。

验证路由器与 PC 之间的连通性。

```
[R1]ping 192.168.2.253
  PING 192.168.2.253: 56  data bytes, press CTRL_C to break
    Reply from 192.168.2.253: bytes=56 Sequence=1 ttl=128 time=500 ms
    Reply from 192.168.2.253: bytes=56 Sequence=2 ttl=128 time=180 ms
    Reply from 192.168.2.253: bytes=56 Sequence=3 ttl=128 time=130 ms
    Reply from 192.168.2.253: bytes=56 Sequence=4 ttl=128 time=90 ms
    Reply from 192.168.2.253: bytes=56 Sequence=5 ttl=128 time=110 ms
  --- 192.168.2.253 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 90/202/500 ms
```

可以正常通信。

思考

请问DHCP服务器在分配地址时是从该网段中最小的地址进行分配还是最大的地址进行分配？这样有什么好处？

12.3 配置DHCP中继

原理概述

由于在IP地址动态获取的过程中，客户端采用广播方式发送请求报文，而广播报文不能跨网段传送，因此DHCP只适用于DHCP客户端和服务器处于同一个网段内的情况。当多个网段都需要进行动态IP地址分配时，就需要在所有网段上都设置一个DHCP服务器，这显然是不易管理和维护的。

DHCP中继可以使客户端通过它与其他网段的DHCP服务器通信，最终获取IP地址，解决了DHCP客户端不能跨网段向服务器动态获取IP地址的问题。这样，在多个不同网络上的DHCP客户端可以使用同一个DHCP服务器，既节省了成本，又便于进行集中管理和维护。路由器或三层交换机都可以充当DHCP中继设备。

实验目的

- 理解DHCP中继的应用场景
- 掌握DHCP中继的配置

实验内容

本实验模拟企业网络场景。某公司分部的网络由交换机S1和网关路由器R1组成，员工终端PC-1和PC-2都连接在S1上。公司要求分部内所有员工主机的IP地址都通过总部的DHCP服务器自动获取。分部网关路由器R1通过公网路由器R2访问公司总部的DHCP服务器R3。由于公司分部与总部不在同一个子网，需要在R1上配置DHCP中继，使分部内主机能跨网段从总部的DHCP服务器自动获取IP地址。

实验拓扑

配置 DHCP 中继的拓扑如图 12-10 所示。

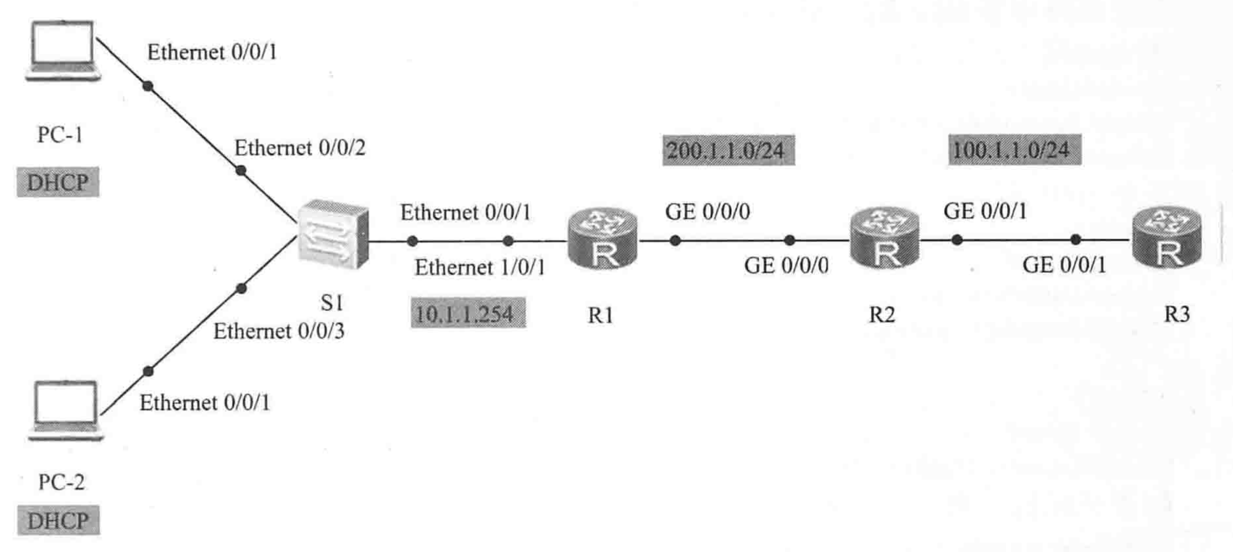


图 12-10 配置 DHCP 中继拓扑

实验编址

实验编址见表 12-3。

表 12-3 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR1220）	Ethernet 1/0/1	10.1.1.254	255.255.255.0	N/A
	GE 0/0/0	200.1.1.1	255.255.255.0	N/A
R2（AR1220）	GE 0/0/0	200.1.1.2	255.255.255.0	N/A
	GE 0/0/1	100.1.1.2	255.255.255.0	N/A
R3（AR1220）	GE 0/0/1	100.1.1.1	255.255.255.0	N/A

实验步骤

1. 基本配置

根据实验编址表进行相应的基本 IP 地址配置，并使用 **ping** 命令检测各直连链路的连通性。

```
<R1>ping 200.1.1.2
  PING 200.1.1.2: 56   data bytes, press CTRL_C to break
    Reply from 200.1.1.2: bytes=56 Sequence=1 ttl=255 time=40 ms
    Reply from 200.1.1.2: bytes=56 Sequence=2 ttl=255 time=30 ms
    Reply from 200.1.1.2: bytes=56 Sequence=3 ttl=255 time=50 ms
    Reply from 200.1.1.2: bytes=56 Sequence=4 ttl=255 time=20 ms
    Reply from 200.1.1.2: bytes=56 Sequence=5 ttl=255 time=50 ms
  --- 200.1.1.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
```

```
0.00% packet loss
round-trip min/avg/max = 20/38/50 ms
```

其余直连网段的连通性测试省略。

2. 搭建 OSPF 网络

在公司路由器 R1、R2、R3 上都配置运行 OSPF 协议，所有网段都发布到区域 0 中。

```
[R1]ospf 1
[R1-ospf-1]area 0
[R1-ospf-1-area-0.0.0.0]network 200.1.1.0 0.0.0.255
[R1-ospf-1-area-0.0.0.0]network 10.1.1.0 0.0.0.255

[R2]ospf1
[R2-ospf-1]area 0
[R2-ospf-1-area-0.0.0.0]network 200.1.1.0 0.0.0.255
[R2-ospf-1-area-0.0.0.0]network 100.1.1.0 0.0.0.255

[R3]ospf 1
[R3-ospf-1]area 0
[R3-ospf-1-area-0.0.0.0]network 100.1.1.0 0.0.0.255
```

配置完成后，查看路由表信息。

```
<R1>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7        Routes : 7
Destination/Mask    Proto    Pre  Cost      Flags NextHop        Interface
10.1.1.0/24         Direct   0    0         D    10.1.1.254      Ethernet1/0/1
10.1.1.254/32       Direct   0    0         D    127.0.0.1       Ethernet1/0/1
100.1.1.0/24        OSPF     10   0         D    200.1.1.2       GigabitEthernet0/0/0
127.0.0.0/8         Direct   0    0         D    127.0.0.1       InLoopBack0
127.0.0.1/32        Direct   0    0         D    127.0.0.1       InLoopBack0
200.1.1.0/24        Direct   0    0         D    200.1.1.1       GigabitEthernet0/0/0
200.1.1.1/32        Direct   0    0         D    127.0.0.1       GigabitEthernet0/0/0

<R2>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7        Routes : 7
Destination/Mask    Proto    Pre  Cost      Flags NextHop        Interface
10.1.1.0/24         OSPF     10   2         D    200.1.1.1       GigabitEthernet0/0/0
100.1.1.0/24        Direct   0    0         D    100.1.1.2       GigabitEthernet0/0/1
100.1.1.2/32        Direct   0    0         D    127.0.0.1       GigabitEthernet0/0/1
127.0.0.0/8         Direct   0    0         D    127.0.0.1       InLoopBack0
127.0.0.1/32        Direct   0    0         D    127.0.0.1       InLoopBack0
200.1.1.0/24        Direct   0    0         D    200.1.1.2       GigabitEthernet0/0/1
200.1.1.2/32        Direct   0    0         D    127.0.0.1       GigabitEthernet0/0/0

<R3>display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 6        Routes : 6
Destination/Mask    Proto    Pre  Cost      Flags NextHop        Interface
```

```
10.1.1.0/24      OSPF    10   3    D  100.1.1.2   GigabitEthernet0/0/1
100.1.1.0/24     Direct  0    0    D  100.1.1.1   GigabitEthernet0/0/1
100.1.1.1/32     Direct  0    0    D  127.0.0.1   GigabitEthernet0/0/1
127.0.0.0/8      Direct  0    0    D  127.0.0.1   InLoopBack0
127.0.0.1/32     Direct  0    0    D  127.0.0.1   InLoopBack0
200.1.1.0/24     OSPF    10   2    D  100.1.1.2   GigabitEthernet0/0/1
```

可以观察到，目前每台设备都可以正常获得路由信息，连通性测试省略。

3. 配置 DHCP 服务器

总部路由器 R3 配置为 DHCP 服务器，负责为分部的网络分配 IP 地址。在 R3 上使用 **dhcp enable** 命令开启 DHCP 功能，创建全局地址池 dhcp-pool，可分配 IP 地址范围为 10.1.1.0/24，出口网关地址为 10.1.1.254。并在面向 DHCP 中继设备的接口上开启 DHCP 服务功能，指定从全局地址池分配地址。

```
[R3]dhcp enable
Info: The operation may take a few seconds. Please wait for a moment.done.
[R3]ip pool dhcp-pool
Info:It's successful to create an IP address pool.
[R3-ip-pool-dhcp-pool]network 10.1.1.0 mask 255.255.255.0
[R3-ip-pool-dhcp-pool]gateway-list 10.1.1.254
[R3-ip-pool-dhcp-pool]interface GigabitEthernet0/0/1
[R3-GigabitEthernet0/0/1]dhcp select global
```

配置完成后，使用 **display ip pool** 命令查看 IP 地址池的配置情况。

```
[R3]display ip pool
-----------------------------------------------------------------------------
Pool-name       : dhcp-pool
Pool-No         : 0
Position        : Local           Status          : Unlocked
Gateway-0       : 10.1.1.254
Mask            : 255.255.255.0
VPN instance    : --
IP address Statistic
Total       :253
Used        :0         Idle        :253
Expired     :0         Conflict    :0          Disable    :0
```

可以观察到，当前可用的地址除去网关 IP 以外还剩下 253 个可用，目前还没有 PC 动态申请 IP 地址。

4. 配置 DHCP 中继

下面配置 R1 为 DHCP 中继设备，指定 DHCP 服务器为 R3。这时如果 R1 从 E 0/0/1 接口收到 PC 的 DHCP 广播请求包，R1 作为 DHCP 中继设备会以单播形式转发请求包到中继所指明的 DHCP 服务器 R3；R3 收到 DHCP 请求包后，会把分配的 IP 地址通过单播包返回给 R1；R1 再把地址信息发送给 PC。

配置指定 DHCP 服务器有两种方式，一种方式是在面向 PC 的接口下直接配置 DHCP 服务器 IP 地址；另一种方式是在面向 PC 的接口下调用全局定义的 DHCP 服务器组。

（1）第一种配置方法：直接在 R1 的 E 0/0/1 接口下开启 DHCP 中继功能，并直接指定 DHCP 服务器 IP 地址为 100.1.1.1。

```
[R1]dhcp enable
[R1]interface Ethernet1/0/1
[R1-Ethernet1/0/1]dhcp select relay
```

```
[R1-Ethernet1/0/1]dhcp relay server-ip 100.1.1.1
```

（2）第二种配置方法：在 R1 上创建 DHCP 服务器组，指定组名为 dhcp-group，并使用 **dhcp-server** 命令添加远端的 DHCP 服务器地址。接着在 E 1/0/1 接口下开启 DHCP 中继功能并配置所对应的 DHCP 服务器组。

```
[R1]dhcp server group dhcp-group
Info:It's successful to create a DHCP server group.
[R1-dhcp-server-group-dhcp-group]dhcp-server 100.1.1.1
[R1-dhcp-server-group-dhcp-group]interface Ethernet 1/0/1
[R1-Ethernet1/0/1]dhcp select relay
[R1-Ethernet1/0/1]dhcp relay server-select dhcp-group
```

两种方式均能达到同样的配置要求，相比而言，在接口下直接指定 DHCP 服务器 IP 地址的方式较简单。但如果中继设备上有多个接口需要配置 DHCP 中继功能，则要在所有接口上重复同样的配置，产生的配置量较大。这种情况就应该使用服务器组的方式，仅在全局定义一次，在每个接口重复调用即可，当有多个 DHCP 服务器或者服务器 IP 地址需要更改时尤为方便。

5. 配置 PC 获取地址方式为 DHCP

当 DHCP 中继设备 R1 和 DHCP 服务器 R3 配置完成后，且中间链路的连通性也正常的情况下，配置 PC 机使用 DHCP 获取 IP 地址，如图 12-11 所示。PC-2 也使用同样的方式配置使用 DHCP。

图 12-11 PC-1 配置界面

配置完成后，在 PC-1 的“命令行”选项卡中使用“**ipconfig**”命令查看地址获得的情况，如图 12-12 所示。

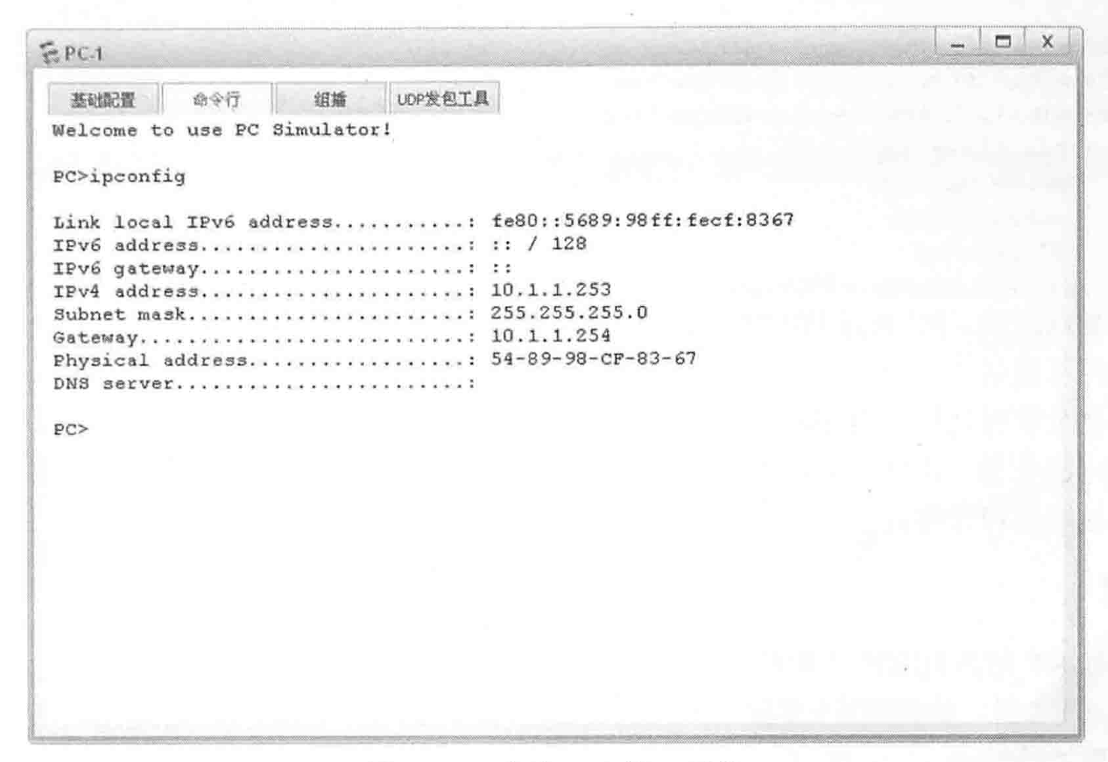


图 12-12 查看 PC-1 的 IP 地址

同样在 PC-2 的“命令行”选项卡中使用**“ipconfig”**命令查看地址获得的情况，如图 12-13 所示。



图 12-13 查看 PC-2 的 IP 地址

现在两台 PC 都从总部 DHCP 服务器获得 IP 地址，测试两台 PC 间的连通性。

```
PC>ping 10.1.1.253
Ping 10.1.1.253: 32 data bytes, Press Ctrl_C to break
From 10.1.1.253: bytes=32 seq=1 ttl=128 time=32 ms
From 10.1.1.253: bytes=32 seq=2 ttl=128 time<1 ms
```

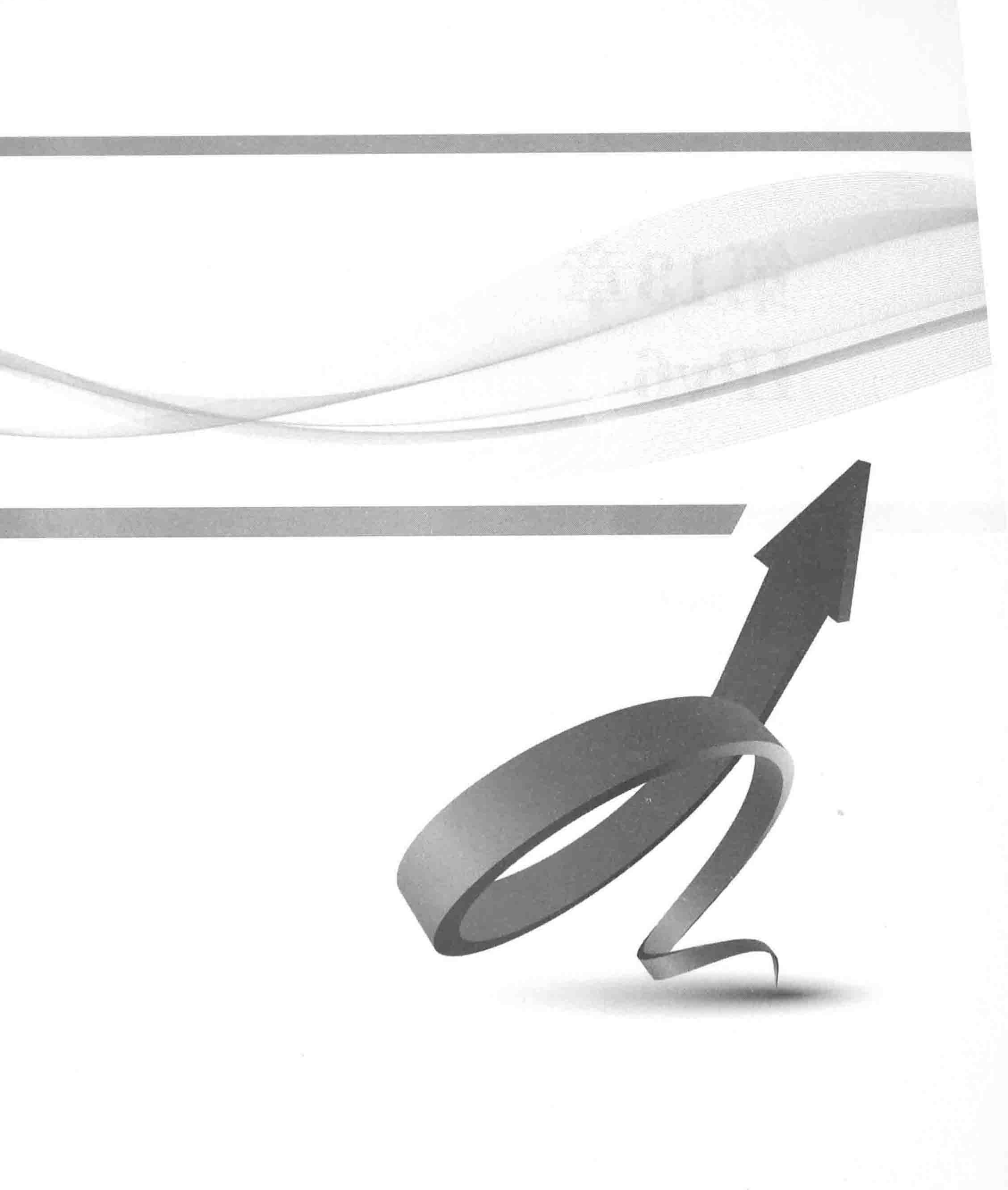
```
From 10.1.1.253: bytes=32 seq=3 ttl=128 time=16 ms
From 10.1.1.253: bytes=32 seq=4 ttl=128 time<1 ms
From 10.1.1.253: bytes=32 seq=5 ttl=128 time=15 ms
--- 10.1.1.253 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 0/12/32 ms
```

测试成功。PC 通过 DHCP 中继设备成功从 DHCP 服务器获得 IP 地址，并能使用该地址相互通信。

整个配置过程，仅在网关路由器 R1 上开启 DHCP 中继功能，在分部没有其他过多的 DHCP 配置。由此可见，在网络设计和管理中灵活使用 DHCP 中继功能能够使网络运行更加高效和方便。

思考

在 R1 充当 DHCP 中继代理时，客户的 DHCP 请求包经 DHCP 中继 R1 到达 DHCP 服务器 R3 后，如果 R3 上定义有不同网段的多个 IP 地址池，R3 如何知道该从哪个地址池分配地址给 PC-1 和 PC-2？

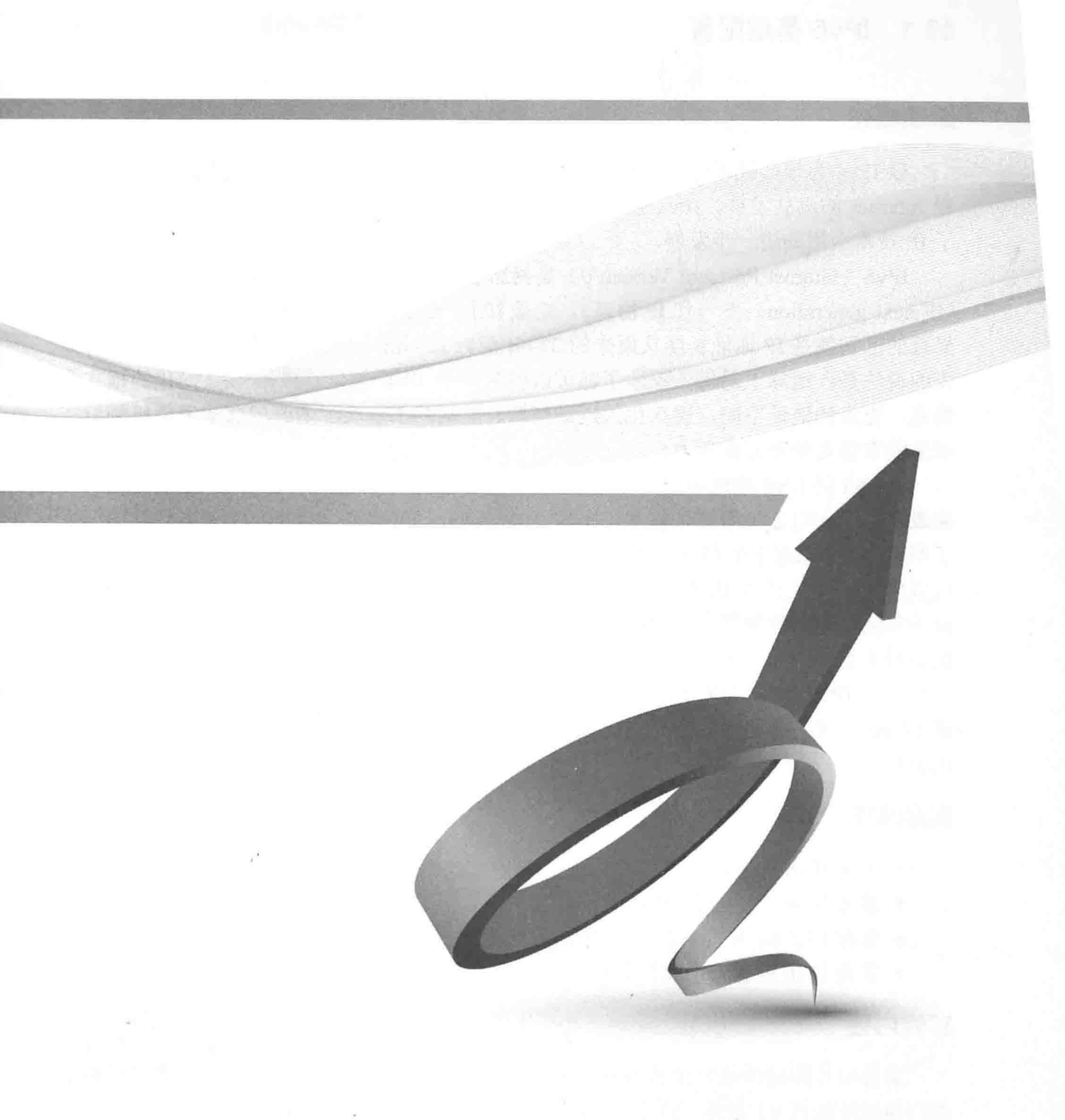


第13章 IPv6

13.1 IPv6基础配置

13.2 RIPng基础配置

13.3 OSPFv3基础配置



13.1 IPv6 基础配置

原理概述

以 IPv4 为核心技术的 Internet 获得巨大成功，促使 IP 技术得到广泛应用。然而，随着 Internet 的迅猛发展，IPv4 技术的不足也日益凸显，特别是地址空间的不足直接限制了 IP 技术应用的进一步发展。

IPv6（Internet Protocol Version 6）是网络层协议的第二代标准协议，也被称为 IPng（IP next generation，下一代 IP 协议）。它是 IETF 设计的一套规范。IPv6 和 IPv4 之间最显著的区别就是 IP 地址长度从原来的 32 bit 变为 128 bit，地址空间大得惊人，有一种夸张的说法是：地球上的每一粒沙子都可以拥有一个 IPv6 地址。IPv6 以其简化的报文头格式、充足的地址空间、层次化的地址结构、灵活的扩展头、增强的邻居发现机制将在未来的市场竞争中充满活力。

128 bit 的 IPv6 地址被分为 8 组，每组的 16 bit 用 4 个十六进制字符（0～9，A～F）来表示，组和组之间用冒号隔开。比如 2031:0000:130F:0000:0000:09C0:876A:130B，为了书写方便，每组中的前导“0”都可以省略。地址中包含的连续两个或多个均为 0 的组，可以用双冒号“::”来代替，这样可以压缩 IPv6 地址书写时的长度。但是在一个 IPv6 地址中只能使用一次双冒号“::”，否则当计算机将压缩后的地址恢复成 128 bit 时，无法确定每段中 0 的个数。所以，上述地址可以简写为 2031:0:130F::9C0:876A:130B。

一个 IPv6 地址可以分为两部分，比如 2001:A304:6101:1:0000:E0:F726:4E58 /64，前 64 bit 是网络前缀，相当于 IPv4 地址中的网络 ID，后 64 bit 相当于 IPv4 地址中的主机 ID。

实验目的

- 理解 IPv6 的地址格式
- 掌握 IPv6 手工配置 IP 地址的方法
- 掌握 EUI-64 方式配置 IPv6 地址的方法
- 掌握 IPv6 静态路由和默认路由的配置方法

实验内容

某公司在新建网络时部署 IPv6，R1 和 R2 分别为 IT 部门和人事部门路由器，两个部门通过交换机 S1 相连。IT 部门的员工终端 PC-1 手工配置 IPv6 地址，并在 R1 与 R2 上配置 IPv6 静态路由，使两个部门的终端能够互相通信。

实验拓扑

IPv6 基础配置的拓扑如图 13-1 所示。

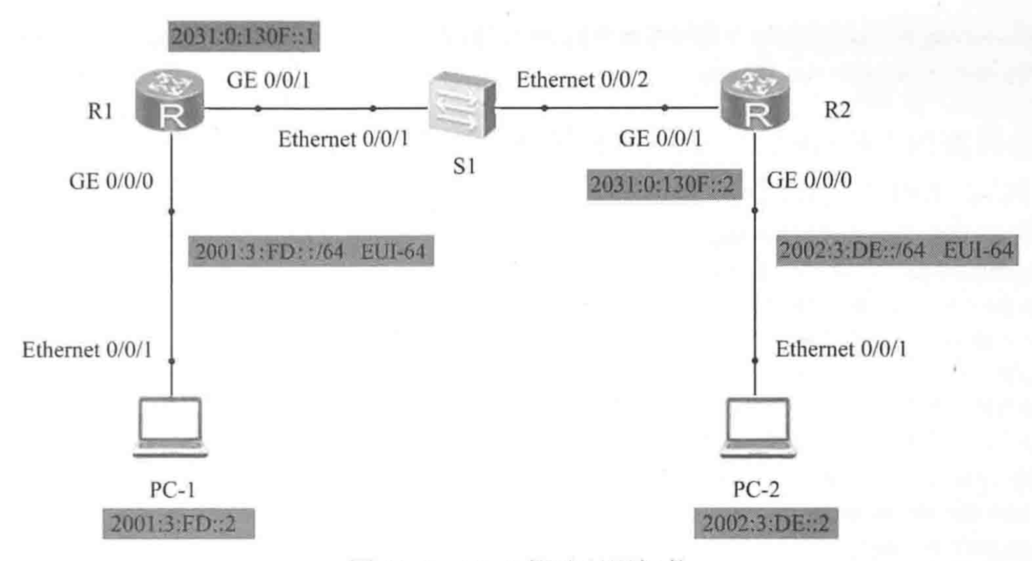


图 13-1　IPv6 基础配置拓扑

实验编址

实验编址见表 13-1。

表 13-1　　实验编址

设备	接口	IPv6 地址	子网掩码	默认网关
R1（AR1220）	GE 0/0/0	2001:3:FD::/64 eui-64	64	N/A
	GE 0/0/1	2031:0:130F::1	64	N/A
R2（AR1220）	GE 0/0/0	2002:3:DE::/64 eui-64	64	N/A
	GE 0/0/1	2031:0:130F::2	64	N/A
PC-1	Ethernet 0/0/1	2001:3:FD::2	64	R1
PC-2	Ethernet 0/0/1	2002:3:DE::2	64	R2

实验步骤

1. 配置 IPv6 单播地址

根据实验编址表在 PC 上配置相应的 IPv6 地址。模拟器中的 PC 上已经默认开启了 IPv6 功能，即已经自动生成了链路本地地址。

在路由器系统视图模式下全局开启 IPv6 功能。

```
<R1>system-view
[R1]ipv6
```

在 R1 上的 GE 0/0/0 接口下使用 **ipv6 enable** 命令开启 IPv6 功能。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]ipv6 enable
```

在 R1 的 GE 0/0/0 接口上配置自动生成的链路本地地址。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]ipv6 address auto link-local
```

配置完成后，在 R1 上查看 GE 0/0/0 接口所配置的自动生成的链路本地地址。

```
[R1]display ipv6 interface
GigabitEthernet0/0/0 current state : UP
IPv6 protocol current state : UP
```

```
IPv6 is enabled, link-local address is FE80::2E0:FCFF:FE03:19AB
  No global unicast address configured
……
```

可以观察到当前 GE 0/0/0 接口下的链路本地地址为 FE80::2E0:FCFF:FE03:19AB。

在 PC-1 上测试与 R1 链路本地地址间的连通性。

```
PC>ping FE80::2E0:FCFF:FE03:19AB
Ping fe80::2e0:fcff:fe03:19ab: 32 data bytes, Press Ctrl_C to break
From fe80::2e0:fcff:fe03:19ab: bytes=32 seq=1 hop limit=64 time=141 ms
From fe80::2e0:fcff:fe03:19ab: bytes=32 seq=2 hop limit=64 time=47 ms
From fe80::2e0:fcff:fe03:19ab: bytes=32 seq=3 hop limit=64 time=47 ms
From fe80::2e0:fcff:fe03:19ab: bytes=32 seq=4 hop limit=64 time=31 ms
From fe80::2e0:fcff:fe03:19ab: bytes=32 seq=5 hop limit=64 time=16 ms
--- fe80::2e0:fcff:fe03:19ab ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 16/56/141 ms
```

通信正常。同理配置 R2 的 GE 0/0/0 接口。

```
[R2]ipv6
[R2]interface GigabitEthernet 0/0/0
[R2-GigabitEthernet0/0/0]ipv6 enable
[R2-GigabitEthernet0/0/0]ipv6 address auto link-local
```

配置完成后，测试 PC-2 与 R2 之间的连通性。

```
PC>ping FE80::2E0:FCFF:FE03:3B30
Ping fe80::2e0:fcff:fe03:3b30: 32 data bytes, Press Ctrl_C to break
From fe80::2e0:fcff:fe03:3b30: bytes=32 seq=1 hop limit=64 time=109 ms
From fe80::2e0:fcff:fe03:3b30: bytes=32 seq=2 hop limit=64 time=16 ms
From fe80::2e0:fcff:fe03:3b30: bytes=32 seq=3 hop limit=64 time<1 ms
From fe80::2e0:fcff:fe03:3b30: bytes=32 seq=4 hop limit=64 time<1 ms
From fe80::2e0:fcff:fe03:3b30: bytes=32 seq=5 hop limit=64 time=16 ms
--- fe80::2e0:fcff:fe03:3b30 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 0/28/109 ms
```

可以观察到，通信正常。

在 R1 和 R2 的 GE 0/0/1 接口上手工静态配置全球单播地址。在配置 IPv4 地址时，新地址会替换老地址；而在配置 IPv6 地址时，新地址会被添加，老地址不受影响。使用 **ipv6 address** 命令可以为接口直接添加 IPv6 地址，2031:0:130F::1 为需要配置的 IPv6 地址，64 为此地址的前缀长度。

```
[R1]interface GigabitEthernet 0/0/1
[R1-GigabitEthernet0/0/1]ipv6 enable
[R1-GigabitEthernet0/0/1]ipv6 address 2031:0:130F::1 64

[R2]interface GigabitEthernet 0/0/1
[R2-GigabitEthernet0/0/1]ipv6 enable
[R2-GigabitEthernet0/0/1]ipv6 address 2031:0:130F::2 64
```

配置完成后在 R1 和 R2 上查看所配置的全局地址。

```
[R1]display ipv6 interface GigabitEthernet 0/0/1
GigabitEthernet0/0/1 current state : UP
IPv6 protocol current state : UP
```

```
IPv6 is enabled, link-local address is FE80::2E0:FCFF:FE03:19AC
Global unicast address(es):
2031:0:130F::1, subnet is 2031:0:130F::/64
Joined group address(es):
……

[R2]display ipv6 interface GigabitEthernet 0/0/1
GigabitEthernet0/0/1 current state : UP
IPv6 protocol current state : UP
IPv6 is enabled, link-local address is FE80::2E0:FCFF:FE03:3B31
Global unicast address(es):
2031:0:130F::2, subnet is 2031:0:130F::/64
Joined group address(es):
……
```

可以观察到，IPv6 全球单播地址的配置已经生效。

测试 R1 与 R2 的全球单播地址间的连通性，同时请注意区分全球单播地址和链路本地地址。

```
<R2>ping ipv6 2031:0:130F::1
  PING 2031:0:130F::1 : 56    data bytes, press CTRL_C to break
    Reply from 2031:0:130F::1
    bytes=56 Sequence=1 hop limit=64    time = 340 ms
    Reply from 2031:0:130F::1
    bytes=56 Sequence=2 hop limit=64    time = 70 ms
    Reply from 2031:0:130F::1
    bytes=56 Sequence=3 hop limit=64    time = 50 ms
    Reply from 2031:0:130F::1
    bytes=56 Sequence=4 hop limit=64    time = 50 ms
    Reply from 2031:0:130F::1
    bytes=56 Sequence=5 hop limit=64    time = 30 ms
  --- 2031:0:130F::1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 30/108/340 ms
```

可以观察到，通信正常。

2. 用 EUI－64 方式配置 IPv6 地址

在 R1 的 GE 0/0/0 接口使用 **ipv6 address** 命令配置 EUI-64 地址。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]ipv6 address 2001:3:FD:: 64 eui-64
```

配置完成后，查看配置结果。

```
[R1]display ipv6 interface brief
*down: administratively down
(l): loopback
(s): spoofing
Interface                    Physical              Protocol
GigabitEthernet0/0/0         up                    up
[IPv6 Address] 2001:3:FD:0:2E0:FCFF:FE03:19AB
GigabitEthernet0/0/1         up                    up
[IPv6 Address] 2031:0:130F::1
```

可以观察到，R1 的 GE 0/0/0 接口此时已经生成了有效的 EUI-64 地址。

继续在 R2 上的 GE 0/0/0 接口使用 **ipv6 address eui-64** 命令配置 EUI-64 地址。

```
[R2]interface GigabitEthernet 0/0/0
[R2-GigabitEthernet0/0/0]ipv6 address 2002:3:DE:: 64 eui-64
```

配置完成后，查看配置结果。

```
[R2]display ipv6 interface brief
*down: administratively down
(l): loopback
(s): spoofing
Interface                    Physical           Protocol
GigabitEthernet0/0/0         up                 up
[IPv6 Address] 2002:3:DE:0:2E0:FCFF:FE03:3B30
GigabitEthernet0/0/1         up                 up
[IPv6 Address] 2031:0:130F::2
```

地址生成后，在 PC-1 上配置 R1 的 GE 0/0/0 接口地址为网关地址，在 PC-2 上配置 R2 的 GE 0/0/0 接口地址为网关地址。

3. 配置 IPv6 静态路由和默认路由

在 R1 上使用 **ipv6 route-static** 命令配置 IPv6 静态路由，目的网段为 PC-2 所在的 IPv6 网段，下一跳为 R2 的 GE 0/0/1 接口的 IPv6 全球单播地址。

```
[R1]ipv6 route-static 2002:3:DE:: 64 2031:0:130F::2
```

在 R2 上配置 IPv6 默认路由，下一跳为 R1 的 GE 0/0/1 接口的 IPv6 全球单播地址。

```
[R2]ipv6 route-static :: 0 2031:0:130F::1
```

配置完成后在 PC-1 上测试与 PC-2 间的连通性。

```
PC>ping 2002:3:DE::2
Ping 2002:3:de::2: 32 data bytes, Press Ctrl_C to break
From 2002:3:de::2: bytes=32 seq=1 hop limit=253 time=31 ms
From 2002:3:de::2: bytes=32 seq=2 hop limit=253 time=47 ms
From 2002:3:de::2: bytes=32 seq=3 hop limit=253 time=47 ms
From 2002:3:de::2: bytes=32 seq=4 hop limit=253 time=47 ms
From 2002:3:de::2: bytes=32 seq=5 hop limit=253 time=32 ms
--- 2002:3:de::2 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 31/40/47 ms
```

可以观察到，通信正常。

思考

如果路由器的某一接口下配置了多个 IPv6 地址，互相之间是否会产生影响？

13.2 RIPng 基础配置

原理概述

RIPng（RIP next generation，下一代 RIP 协议）是 IPv4 中 RIPv 2 协议在 IPv6 网络

上的扩展，多数 RIPv2 的原理都可以适用于 RIPng。RIPng 协议同样是基于距离矢量算法的路由协议，用跳数来衡量到达目的主机的距离（也称为度量值或开销）。在 RIPng 协议中，当跳数大于或等于 16 时，目的网络或主机就被定义为不可达。

为了能在 IPv6 网络中应用，RIPng 对原有的 RIP 协议进行了修改。

■ UDP 端口号：使用 UDP 的 521 端口（RIP 使用 520 端口）发送和接收路由信息；

■ 组播地址：使用 FF02::9 作为链路本地范围内的 RIPng 路由器组播地址；

■ 目的地址和下一跳地址：使用 128 bit 的 IPv6 地址，并使用前缀长度来代替子网掩码。

RIPng 协议路由算法和 RIPv2 一样，同样支持水平分割、毒性逆转和触发更新功能，用来防止环路。默认情况下，启用水平分割功能和触发更新，不启用毒性逆转功能。

实验目的

- 理解 RIPng 的应用场景
- 掌握 RIPng 的配置
- 理解 RIPng 配置与 RIP 配置的区别

实验内容

某公司内部网络是个小型的 IPv6 网络，公司内 IT 部门通过路由器 R2 与公司出口网关 R1 相连，人事部门通过路由器 R3 与网关 R1 相连。由于公司网络是个简单的网络，本实验通过配置 RIPng 协议使各 IPv6 网络互通。

实验拓扑

RIPng 基础配置的拓扑如图 13-2 所示。

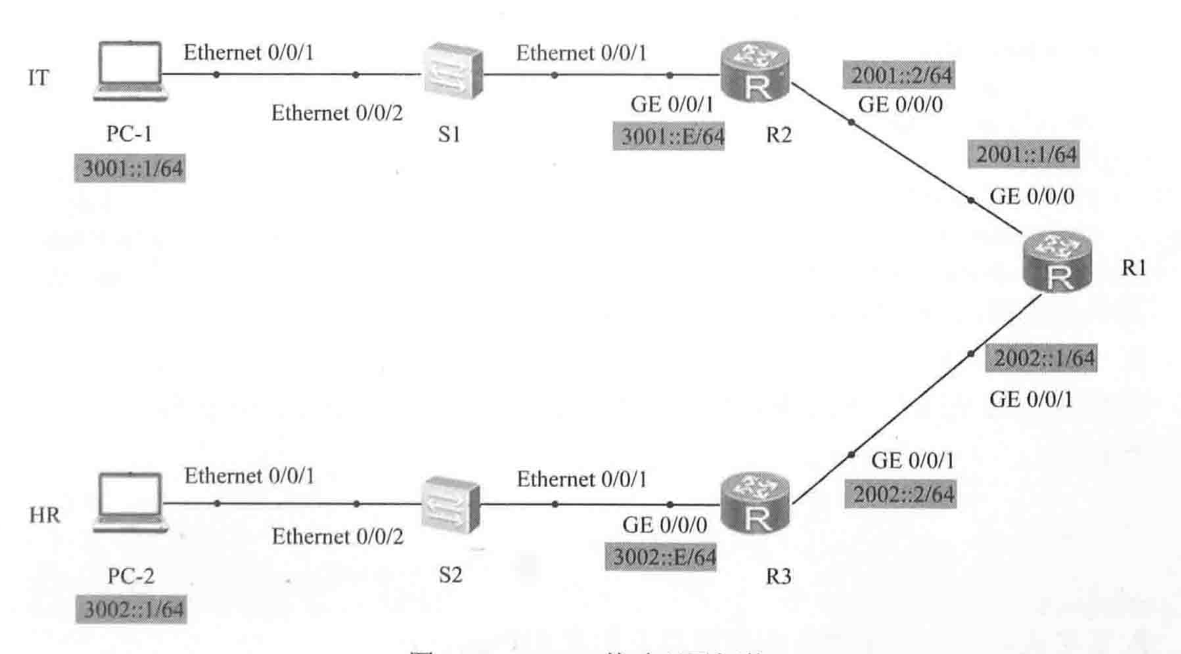


图 13-2 RIPng 基础配置拓扑

实验编址

实验编址见表 13-2。

表 13-2 实验编址

设备	接口	IPv6 地址	子网掩码	默认网关
R1（AR2220）	GE 0/0/0	2001::1	64	N/A
	GE 0/0/1	2002::1	64	N/A
R2（AR2220）	GE 0/0/1	3001::E	64	N/A
	GE 0/0/0	2001::2	64	N/A
R3（AR2220）	GE 0/0/0	3002::E	64	N/A
	GE 0/0/1	2002::2	64	N/A
PC-1	Ethernet 0/0/1	3001::1	64	3001::E
PC-2	Ethernet 0/0/1	3002::1	64	3002::E

实验步骤

1. 基本配置

根据实验编址表配置各接口的 IPv6 地址，并使用 **ping** 命令检测各直连链路的连通性。

```
[R1]ping ipv6 2002::2
  PING 2002::2 : 56    data bytes, press CTRL_C to break
    Reply from 2002::2
    bytes=56 Sequence=1 hop limit=64    time = 140 ms
    Reply from 2002::2
    bytes=56 Sequence=2 hop limit=64    time = 50 ms
    Reply from 2002::2
    bytes=56 Sequence=3 hop limit=64    time = 70 ms
    Reply from 2002::2
    bytes=56 Sequence=4 hop limit=64    time = 30 ms
    Reply from 2002::2
    bytes=56 Sequence=5 hop limit=64    time = 20 ms
  --- 2002::2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 20/62/140 ms
```

其余直连网段的连通性测试省略。

2. 配置 RIPng

根据图 13-2 在 R1、R2、R3 上配置 RIPng 协议，创建 RIPng 路由进程 1。

```
[R1]ripng 1

[R2]ripng 1

[R3]ripng 1
```

配置完成后，在路由器各相应接口下配置 RIPng。

```
[R1]interface GigabitEthernet 0/0/0
```

```
[R1-GigabitEthernet0/0/0]ripng 1 enable
[R1-GigabitEthernet0/0/0]interface GigabitEthernet 0/0/1
[R1-GigabitEthernet0/0/1]ripng 1 enable

[R2]interface GigabitEthernet 0/0/0
[R2-GigabitEthernet0/0/0]ripng 1 enable
[R2-GigabitEthernet0/0/0]interface GigabitEthernet 0/0/1
[R2-GigabitEthernet0/0/1]ripng 1 enable

[R3]interface GigabitEthernet 0/0/0
[R3-GigabitEthernet0/0/0]ripng 1 enable
[R3-GigabitEthernet0/0/0]interface GigabitEthernet 0/0/1
[R3-GigabitEthernet0/0/1]ripng 1 enable
```

3. 检查RIPng的路由表

配置完成后，查看每台路由器RIPng的路由表。

```
[R1]display ipv6 routing-table
Routing Table : Public
Destinations : 8          Routes : 10
......
 Destination   : 3001::                        PrefixLength : 64
 NextHop       : FE80::5689:98FF:FE76:832B     Preference   : 100
 Cost          : 1                             Protocol     : RIPng
 RelayNextHop  : ::                            TunnelID     : 0x0
 Interface     : GigabitEthernet0/0/0          Flags        : D
......
 Destination   : 3002::                        PrefixLength : 64
 NextHop       : FE80::5689:98FF:FE76:8224     Preference   : 100
 Cost          : 1                             Protocol     : RIPng
 RelayNextHop  : ::                            TunnelID     : 0x0
 Interface     : GigabitEthernet0/0/1          Flags        : D
......

[R2]display ipv6 routing-table
Routing Table : Public
Destinations : 8          Routes : 12
......
Destination    : 2002::                        PrefixLength : 64
 NextHop       : FE80::5689:98FF:FE76:8223     Preference   : 100
 Cost          : 1                             Protocol     : RIPng
 RelayNextHop  : ::                            TunnelID     : 0x0
 Interface     : GigabitEthernet0/0/0          Flags        : D
......
 Destination   : 3002::                        PrefixLength : 64
 NextHop       : FE80::5689:98FF:FE76:8223     Preference   : 100
 Cost          : 1                             Protocol     : RIPng
 RelayNextHop  : ::                            TunnelID     : 0x0
 Interface     : GigabitEthernet0/0/0          Flags        : D
......

[R3]display ipv6 routing-table
```

```
Routing Table : Public
Destinations : 8        Routes : 12
......
Destination   : 2001::                          PrefixLength : 64
 NextHop        : FE80::5689:98FF:FE76:830C      Preference    : 100
 Cost           : 1                              Protocol      : RIPng
 RelayNextHop : ::                              TunnelID      : 0x0
 Interface      : GigabitEthernet0/0/1           Flags         : D
......
 Destination   : 3001::                          PrefixLength : 64
 NextHop        : FE80::5689:98FF:FE76:830C      Preference    : 100
 Cost           : 1                              Protocol      : RIPng
 RelayNextHop : ::                              TunnelID      : 0x0
 Interface      : GigabitEthernet0/0/1           Flags         : D
......
```

可以观察到各个路由器都获取了相应的 RIPng 路由信息。

在路由器 R1、R2、R3 使用 **display ripng 1 route** 命令查看 RIPng1 的路由信息，同样也可以观察到每台路由器上获取到的 RIPng 路由信息，以 R1 为例。

```
[R1]display ripng 1 route
   Route Flags: R - RIPng
                A - Aging, G - Garbage-collect
------------------------------------------------------------------------
 Peer FE80::5689:98FF:FE76:832B on GigabitEthernet0/0/0
 Dest 3001::/64,
      via FE80::5689:98FF:FE76:832B, cost  1, tag 0, RA, 29 s
 Peer FE80::5689:98FF:FE76:832C on GigabitEthernet0/0/1
 Dest 3001::/64,
      via FE80::5689:98FF:FE76:832B, cost  1, tag 0, RA, 14 s
 Peer FE80::5689:98FF:FE76:8223 on GigabitEthernet0/0/0
 Dest 3002::/64,
      via FE80::5689:98FF:FE76:8224, cost  1, tag 0, RA, 15 s
 Peer FE80::5689:98FF:FE76:8224 on GigabitEthernet0/0/1
 Dest 3002::/64,
      via FE80::5689:98FF:FE76:8224, cost  1, tag 0, RA, 13 s
```

在 PC-1 上测试与 PC-2 间的连通性。

```
PC>ping 3002::1
Ping 3002::1: 32 data bytes, Press Ctrl_C to break
From 3002::1: bytes=32 seq=1 hop limit=64 time=172 ms
From 3002::1: bytes=32 seq=2 hop limit=64 time=46 ms
From 3002::1: bytes=32 seq=3 hop limit=64 time=93 ms
From 3002::1: bytes=32 seq=4 hop limit=64 time=16 ms
From 3002::1: bytes=32 seq=5 hop limit=64 time=16 ms
--- 3002::1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 16/68/172 ms
```

可以观察到通信正常，RIPng 协议已经使全网互通。

思考

RIPng 支持认证吗？为什么？

13.3 OSPFv3 基础配置

原理概述

OSPF 针对 IPv4 协议使用的是 Version 2，针对 IPv6 协议使用的是 Version 3，即 OSPFv3。OSPFv3 在 OSPFv2 基础上进行了增强，是一种运行在 IPv6 网络之上的路由协议。

OSPFv2 是基于 IPv4 子网运行的，同一链路上的所有节点同处于一个 IPv4 子网或网络内，邻居关系建立的前提之一是相连接口必须处于同一 IPv4 子网内，每一条路由的下一跳地址都是和路由器接口处于同一网段的 IPv4 地址。OSPFv3 是基于链路运行的，同一链路上的两个节点不必具有相同的前缀也可以直接通信，这一点极大地改变了 OSPF 的行为，使它独立于网络协议，容易扩展适应各种协议。

OSPFv3 的 Router-ID、Area ID 仍然保留类似 IPv4 地址长度的 32 bit 的格式。实际上这些字段既不是 IPv4 地址，也不是 IPv6 地址，而只是一个编号。

另外在 OSPFv2 中，对于 Broadcast 和 NBMA 网络类型，邻居路由器是以 IP 地址作为标识的。而在 OSPFv3 中，邻居路由器总是以 Router-ID 作为标识的，所以 DR 和 BDR 也总是用其 Router-ID 来标识的。

OSPFv3 不再直接提供验证功能，转而依赖 IPv6 所提供的 IP AH（Authentication Header）和 IP ESP（Encapsulating Security Payload）协议进行验证，以确保路由信息的可信性、完整性和机密性。

实验目的

- 理解 OSPFv3 的应用场景
- 掌握 OSPFv3 的配置
- 理解 OSPFv3 配置与 OSPFv2 配置的区别
- 理解 OSPFv3 基于链路运行的特点

实验内容

公司内部网络是个中型的 IPv6 网络，R1 和 R2 是公司两台核心路由器，R3 是 IT 部门路由器，与核心层路由器 R1 直连。R4 为人事部门路由器，与核心层路由器 R2 直连。为了使公司内网所有部门网络能互相通信，需要在此网络中配置支持 IPv6 的动态路由协议。考虑到公司网络较大以及网络的扩展，部署 OSPFv3。核心层路由器之间为区域 0，整个 IT 部门在区域 1 中，人事部在区域 2 中。

实验拓扑

OSPFv3 基础配置的拓扑如图 13-3 所示。

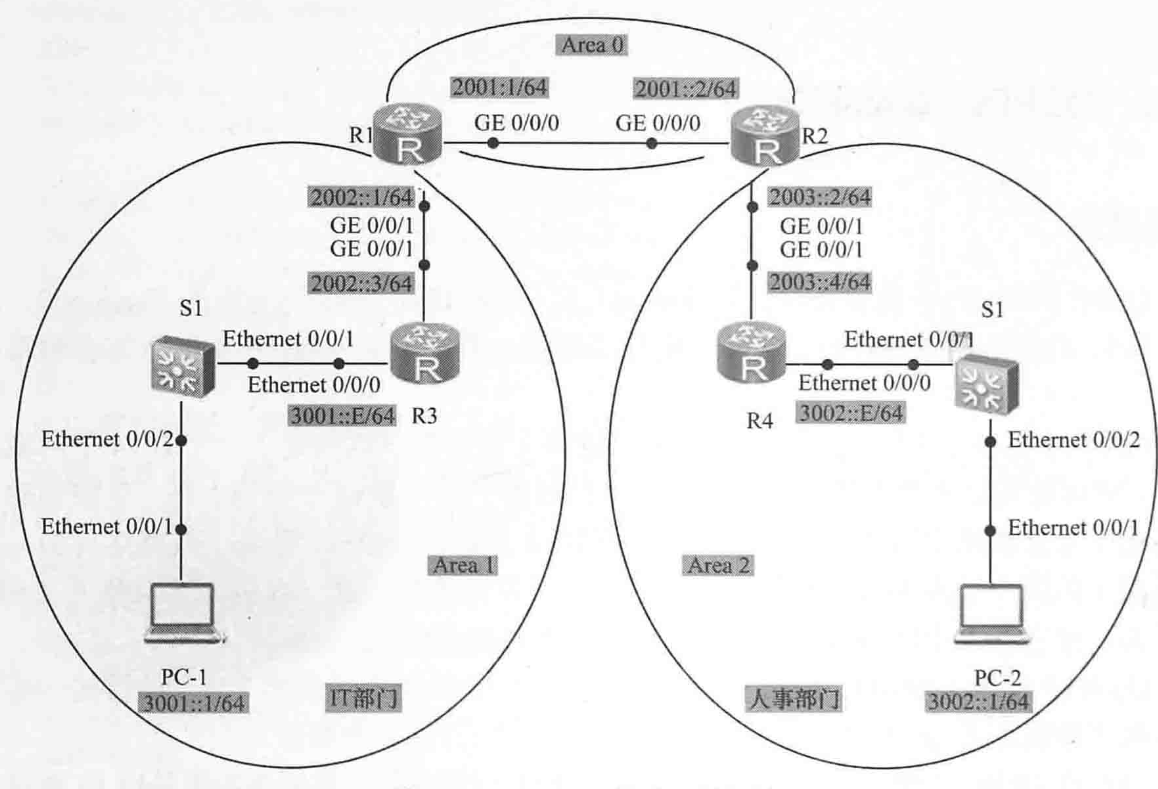


图 13-3　OSPFv3 基础配置拓扑

实验编址

实验编址见表 13-3。

表 13-3　　　　实验编址

设备	接口	IPv6 地址	子网掩码	默认网关
R1（AR1220）	GE 0/0/0	2001::1	64	N/A
	GE 0/0/1	2002::1	64	N/A
R2（AR1220）	GE 0/0/0	2001::2	64	N/A
	GE 0/0/1	2003::2	64	N/A
R3（AR1220）	GE 0/0/1	2002::3	64	N/A
	Ethernet 0/0/0	3001::E	64	N/A
R4（AR1220）	GE 0/0/1	2003::4	64	N/A
	Ethernet 0/0/0	3002::E	64	N/A
PC-1	Ethernet 0/0/1	3001::1	64	3001::E
PC-2	Ethernet 0/0/1	3002::1	64	3002::E

实验步骤

1. 基本配置

在各台设备上开启 IPv6 功能，根据实验编址表进行相应的基本 IPv6 地址配置，并

使用 **ping** 命令检测各直连链路的连通性。

```
[R1]ping ipv6 2001::2
  PING 2001::2 : 56   data bytes, press CTRL_C to break
    Reply from 2001::2
    bytes=56 Sequence=1 hop limit=64   time = 270 ms
    Reply from 2001::2
    bytes=56 Sequence=2 hop limit=64   time = 70 ms
    Reply from 2001::2
    bytes=56 Sequence=3 hop limit=64   time = 50 ms
    Reply from 2001::2
    bytes=56 Sequence=4 hop limit=64   time = 40 ms
    Reply from 2001::2
    bytes=56 Sequence=5 hop limit=64   time = 40 ms
  --- 2001::2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 40/94/270 ms
```

其余直连网段的连通性测试省略。

2. 搭建 OSPFv3 网络

在各路由器上创建 OSPFv3 进程 1，并在 OSPFv3 进程中配置每台路由器的 OSPF Router-ID。R1、R2、R3、R4 的 Router-ID 分别为 1.1.1.1、2.2.2.2、3.3.3.3、4.4.4.4。

```
[R1]ospfv3 1
[R1-ospfv3-1]router-id 1.1.1.1

[R2]ospfv3 1
[R2-ospfv3-1]router-id 2.2.2.2

[R3]ospfv3 1
[R3-ospfv3-1]router-id 3.3.3.3

[R4]ospfv3 1
[R4-ospfv3-1]router-id 4.4.4.4
```

如果不配置 Router-ID，OSPFv3 的邻居就无法建立。因为在 OSPFv3 中，路由器是以 Router-ID 作为标识的，而不是用接口地址来标识。

在接口下配置 OSPFv3，区域按照图 13-3 划分的区域进行配置，IT 部门属于区域 1，人事部门属于区域 2，区域 1 和区域 2 通过骨干区域 0 相连。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]ospfv3 1 area 0

[R1]interface GigabitEthernet 0/0/1
[R1-GigabitEthernet0/0/1]ospfv3 1 area 1

[R2]interface GigabitEthernet 0/0/0
[R2-GigabitEthernet0/0/0]ospfv3 1 area 0

[R2]interface GigabitEthernet 0/0/1
[R2-GigabitEthernet0/0/1]ospfv3 1 area 2
```

```
[R3]interface GigabitEthernet 0/0/1
[R3-GigabitEthernet0/0/1]ospfv3 1 area 1

[R3]interface Ethernet 0/0/0
[R3-Ethernet0/0/0]ospfv3 1 area 1

[R4]interface GigabitEthernet 0/0/1
[R4-GigabitEthernet0/0/1]ospfv3 1 area 2

[R4]interface Ethernet 0/0/0
[R4-Ethernet0/0/0]ospfv3 1 area 2
```

配置完成后，查看每台路由器上的 OSPFv3 邻居状态。

```
[R1]display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.0)
Neighbor ID     Pri  State         Dead Time    Interface     Instance ID
2.2.2.2           1  Full/DR       00:00:37     GE0/0/0                 0
OSPFv3 Area (0.0.0.1)
Neighbor ID     Pri  State         Dead Time    Interface     Instance ID
3.3.3.3           1  Full/DR       00:00:36     GE0/0/1                 0

[R2]display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.0)
Neighbor ID     Pri  State         Dead Time    Interface     Instance ID
1.1.1.1           1  Full/Backup   00:00:32     GE0/0/0                 0
OSPFv3 Area (0.0.0.2)
Neighbor ID     Pri  State         Dead Time    Interface     Instance ID
4.4.4.4           1  Full/DR       00:00:34     GE0/0/1                 0

[R3]display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.1)
Neighbor ID     Pri  State         Dead Time    Interface     Instance ID
1.1.1.1           1  Full/Backup   00:00:34     GE0/0/1                 0

[R4]display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.2)
Neighbor ID     Pri  State         Dead Time    Interface     Instance ID
2.2.2.2           1  Full/Backup   00:00:32     GE0/0/1                 0
```

可以观察到，所有路由器都成功建立起了 OSPFv3 邻居。

查看每台路由器的 IPv6 路由表。

```
[R1]display ipv6 routing-table
Routing Table : Public
        Destinations : 9        Routes : 9
......
Destination   : 3001::                       PrefixLength : 64
```

```
 NextHop        : FE80::2E0:FCFF:FE03:1B68     Preference    : 10
 Cost           : 2                            Protocol      : OSPFv3
 RelayNextHop : ::                             TunnelID      : 0x0
 Interface      : GigabitEthernet0/0/1         Flags         : D
 Destination    : 3002::                       PrefixLength : 64
 NextHop        : FE80::2E0:FCFF:FE03:56F0     Preference    : 10
 Cost           : 3                            Protocol      : OSPFv3
 RelayNextHop : ::                             TunnelID      : 0x0
 Interface      : GigabitEthernet0/0/0         Flags         : D
......

[R2]display ipv6 routing-table
Routing Table : Public
       Destinations : 9       Routes : 9
......
 Destination    : 3001::                       PrefixLength : 64
 NextHop        : FE80::2E0:FCFF:FE03:F661     Preference    : 10
 Cost           : 3                            Protocol      : OSPFv3
 RelayNextHop : ::                             TunnelID      : 0x0
 Interface      : GigabitEthernet0/0/0         Flags         : D
 Destination    : 3002::                       PrefixLength : 64
 NextHop        : FE80::2E0:FCFF:FE03:2906     Preference    : 10
 Cost           : 2                            Protocol      : OSPFv3
 RelayNextHop : ::                             TunnelID      : 0x0
 Interface      : GigabitEthernet0/0/1         Flags         : D
......

[R3]display ipv6 routing-table
Routing Table : Public
       Destinations : 9       Routes : 9
......
Destination    : 3002::                        PrefixLength : 64
 NextHop        : FE80::2E0:FCFF:FE03:F662     Preference    : 10
 Cost           : 4                            Protocol      : OSPFv3
 RelayNextHop : ::                             TunnelID      : 0x0
 Interface      : GigabitEthernet0/0/1         Flags         : D
......

[R4]display ipv6 routing-table
Routing Table : Public
       Destinations : 9       Routes : 9
......
 Destination    : 3001::                       PrefixLength : 64
 NextHop        : FE80::2E0:FCFF:FE03:56F1     Preference    : 10
 Cost           : 4                            Protocol      : OSPFv3
 RelayNextHop : ::                             TunnelID      : 0x0
 Interface      : GigabitEthernet0/0/1         Flags         : D
......
```

可以观察到，各个路由器之间相互接收到了添加进 OSPFv3 进程接口所在网段的路由条目。

在 PC-1 上测试与 PC-2 间的连通性。

```
PC>ping 3002::1
Ping 3002::1: 32 data bytes, Press Ctrl_C to break
From 3002::1: bytes=32 seq=1 hop limit=64 time=47 ms
From 3002::1: bytes=32 seq=2 hop limit=64 time=32 ms
From 3002::1: bytes=32 seq=3 hop limit=64 time=31 ms
From 3002::1: bytes=32 seq=4 hop limit=64 time=31 ms
From 3002::1: bytes=32 seq=5 hop limit=64 time=31 ms
--- 3002::1 ping statistics ---
  5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
  round-trip min/avg/max = 31/34/47 ms
```

此时主机 PC-1 和 PC-2 可以通过 OSPFv3 路由协议进行通信。

3. 验证 OSPFv3 建立邻居的特性

在 R1 的 GE 0/0/0 接口下删除之前配置的 IPv6 地址。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]undo ipv6 address
```

配置完成后，在 R1 上使用 **display ospfv3 peer** 命令查看邻居关系。

```
[R1]display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.1)
Neighbor ID     Pri  State          Dead Time Interface          Instance ID
3.3.3.3           1  Full/DR         00:00:36   GE0/0/1                      0
```

可以观察到，此时 R1 上已经没有与 R2 间的邻居关系。

在 R1 的 GE 0/0/0 接口上配置与 R2 直连接口所在 IPv6 网段不同前缀的 IPv6 地址 2009::1/64。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]ipv6 address 2009::1
```

配置完成后，在 R1 上使用 **display ospfv3 peer** 命令查看邻居关系。

```
[R1]display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.0)
Neighbor ID     Pri  State          Dead Time Interface          Instance ID
2.2.2.2           1  Full/DR         00:00:31   GE0/0/0                      0
OSPFv3 Area (0.0.0.1)
Neighbor ID     Pri  State          Dead Time Interface          Instance ID
3.3.3.3           1  Full/DR         00:00:34   GE0/0/1                      0
```

可以观察到 R1 与 R2 仍然能建立邻居关系。这是因为 OSPFv3 的邻居关系建立是通过 Link-Local 地址来实现的，即链路上两端的 IPv6 地址即使拥有不同的前缀也可以建立邻居关系，而 OSPFv2 在同一链路上必须要使用相同网段的 IPv4 地址才能建立邻居关系。

在 R1 上查看 GE 0/0/0 接口上的 Link-Local 地址。

```
[R1]display ipv6 interface GigabitEthernet 0/0/0
GigabitEthernet0/0/0 current state : UP
IPv6 protocol current state : UP
```

```
IPv6 is enabled, link-local address is FE80::2E0:FCFF:FE03:330F
  Global unicast address(es):
……
```

当通过改变接口的 IPv6 地址时，Link-Local 地址是不会改变的，所以 R1 与 R2 的 OSPFv3 邻居关系仍然可以建立起来。

思考

在本实验中，OSPFv3 协议修改了全球单播地址，邻居关系未受影响，请问各网络间的通信是否正常？从此处能够得到什么启示？

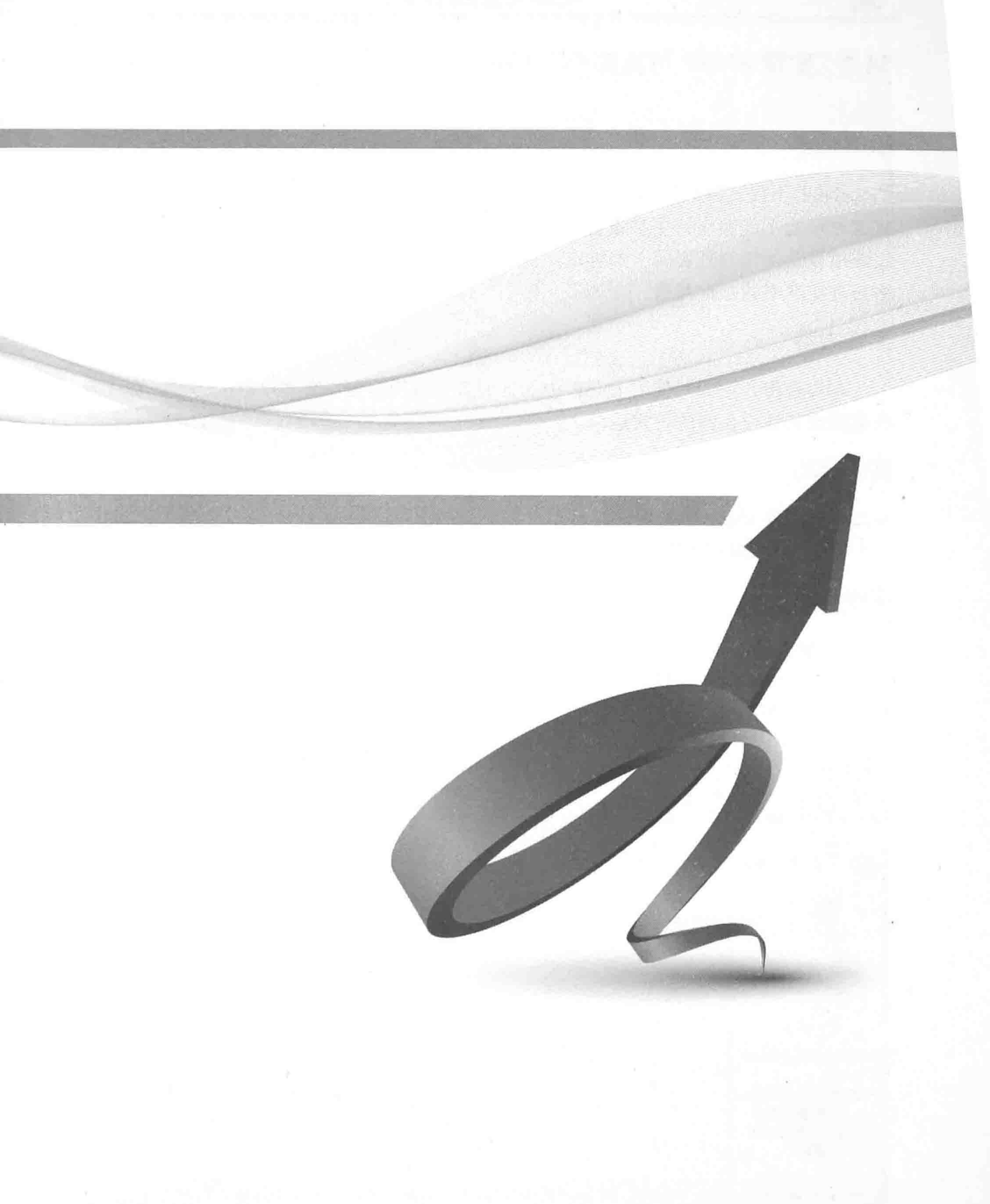
第14章 其他特性

14.1 实现eNSP与真实PC桥接

14.2 SNMP基础配置

14.3 GRE协议基础配置

14.4 配置NAT



14.1 实现 eNSP 与真实 PC 桥接

原理概述

eNSP 不仅支持单机部署，同时还支持 Server 端分布式部署在多台服务器上，分布式部署环境下能够支持更多设备组成复杂的大型网络。eNSP 还可与真实设备对接，通过虚拟设备接口与真实网卡的绑定，实现虚拟设备与真实设备的对接，进而实现虚拟网络与真实网络的互连互通。

实验内容

本实验将介绍如何使用 eNSP 中的云设备实现模拟器与真实电脑的桥接，实现与真实电脑或其他设备间的正常通信。

实验目的

- 掌握在 eNSP 中使用云设备与虚拟 PC 连接的方法
- 掌握在 eNSP 中使用云设备与真实 PC 上的物理网卡桥接的方法

实验步骤

1. 配置云设备

双击 eNSP 图标，打开模拟器，选择界面左侧 Cloud 栏的“云设备”图标，拖进拓扑图中，如图 14-1 所示。



图 14-1 选取设备

在云设备图标上单击鼠标右键，在弹出的快捷菜单中选择“设置”命令，如图 14-2 所示。

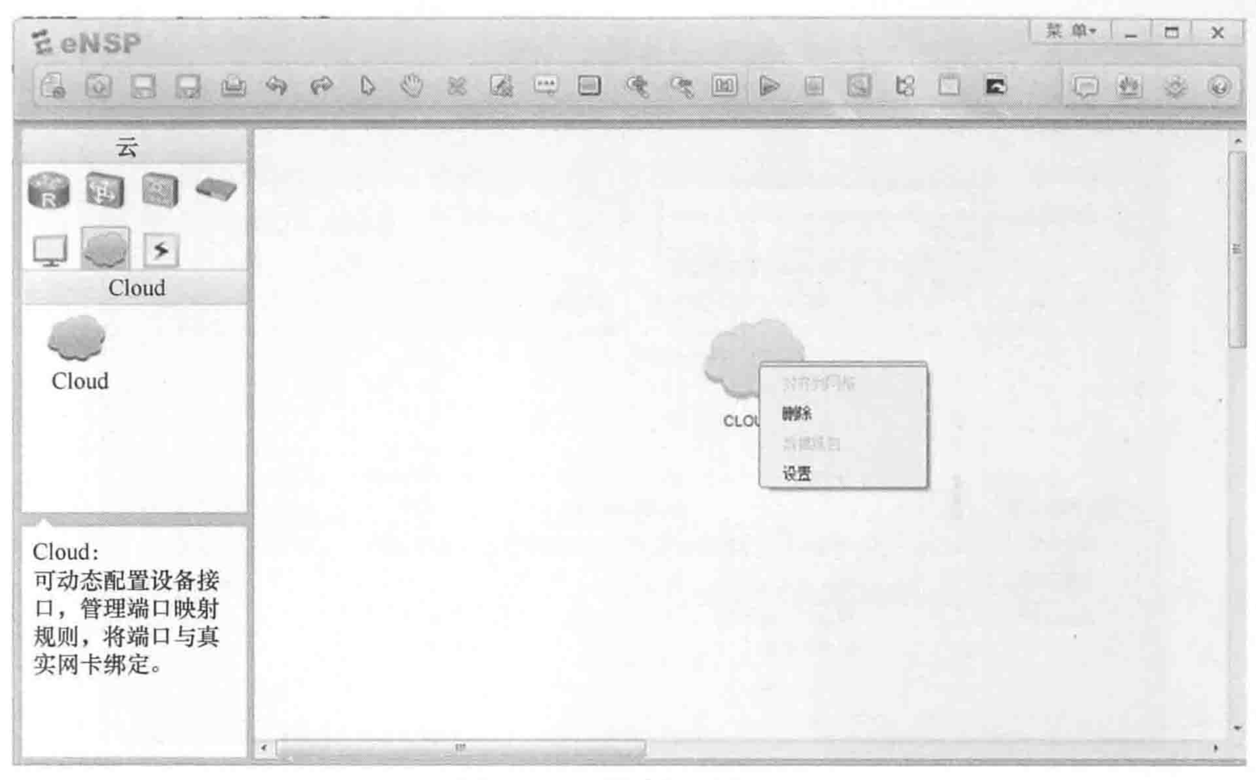


图 14-2 右键选择“设置”

进入云设置界面后，创建一个端口。在“绑定信息”下拉列表中选择“UDP”，在“端口类型”下拉列表中选择“Ethernet”，然后单击“增加”按钮，新创建端口的信息将会出现在端口信息表中，序列号为 1，如图 14-3 所示。

CLOUD1
IO 配置
端口创建
绑定信息： UDP
端口类型： Ethernet 开放UDP端口
监听端口： 30000 (30000-35000)
对端IP： 0 . 0 . 0 . 0
对端端口： 0 修改
增加 删除

No.	端口类型	端口编号	UDP端口号	端口开放状态	绑定信息
1	Ethernet	1	5002	Internal	UDP

端口映射设置
端口类型： Ethernet
入端口编号： 1
出端口编号： 1
双向通道
增加

端口映射表

No.	入端口编号	出端口编号	端口类型

删除

图 14-3 创建端口

创建另外一个端口。“绑定信息”选择真实 PC 中任意一个网卡地址，这里选择了 PC 中的无线网卡，IP 地址为 192.168.6.33，“端口类型”仍然选择“Ethernet”，如图 14-4 所示。

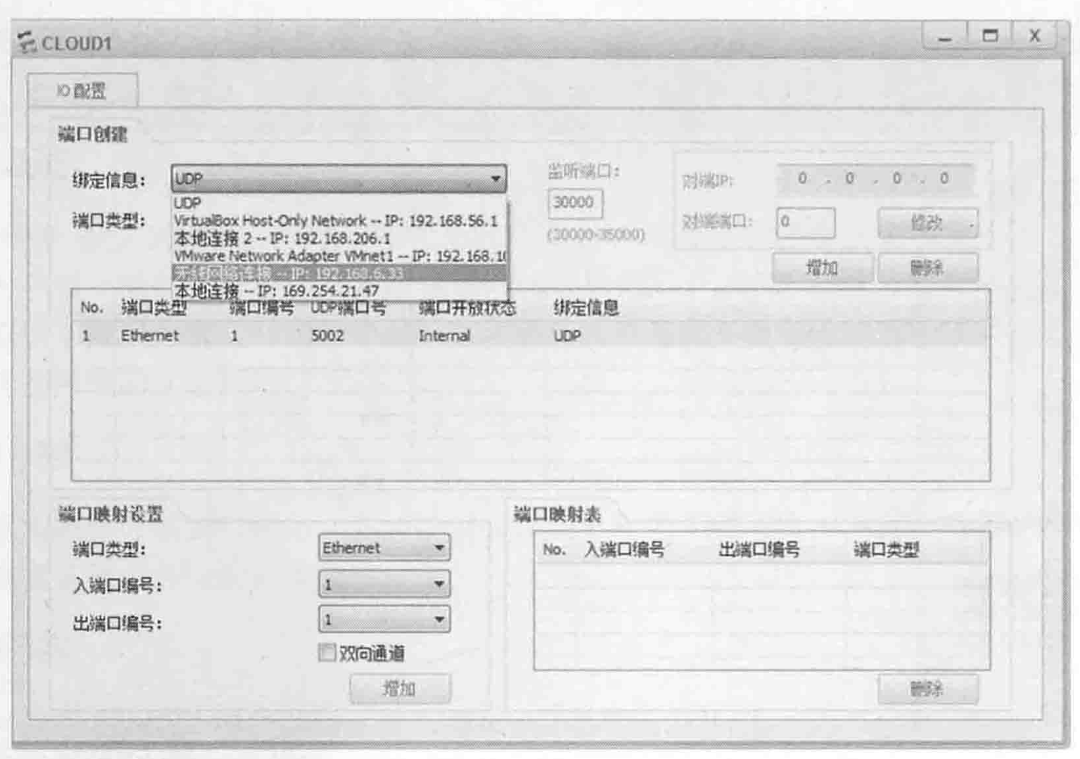


图 14-4　创建端口

单击“增加”按钮，端口列表中会显示第 2 个端口的信息，如图 14-5 所示。

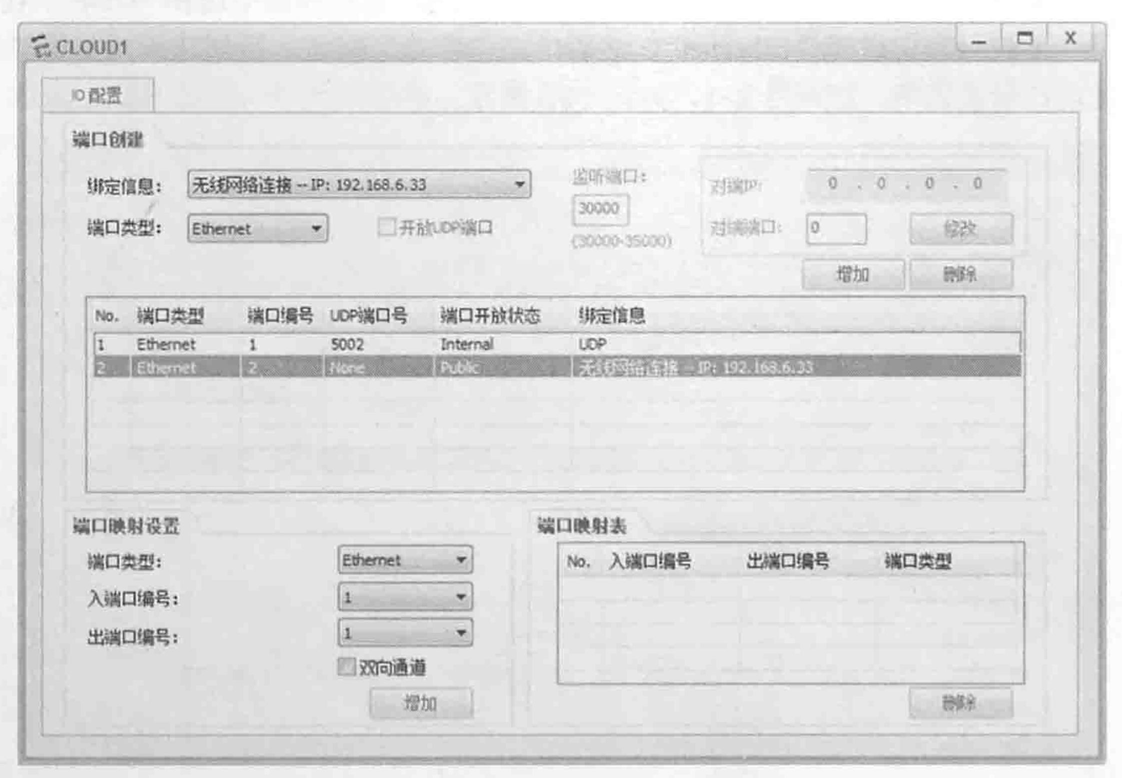


图 14-5　创建端口

接下来在“端口映射设置”栏中创建端口的映射关系。在“入端口编号”下拉列表中选择“2”，也就是刚才所创建的对应真实 PC 上无线网卡的端口；将“出端口编号”选择为“1”，选中“双向通道”复选框，然后单击“增加”按钮，即可添加到右侧的“端口映射表”中，如图 14-6 所示。

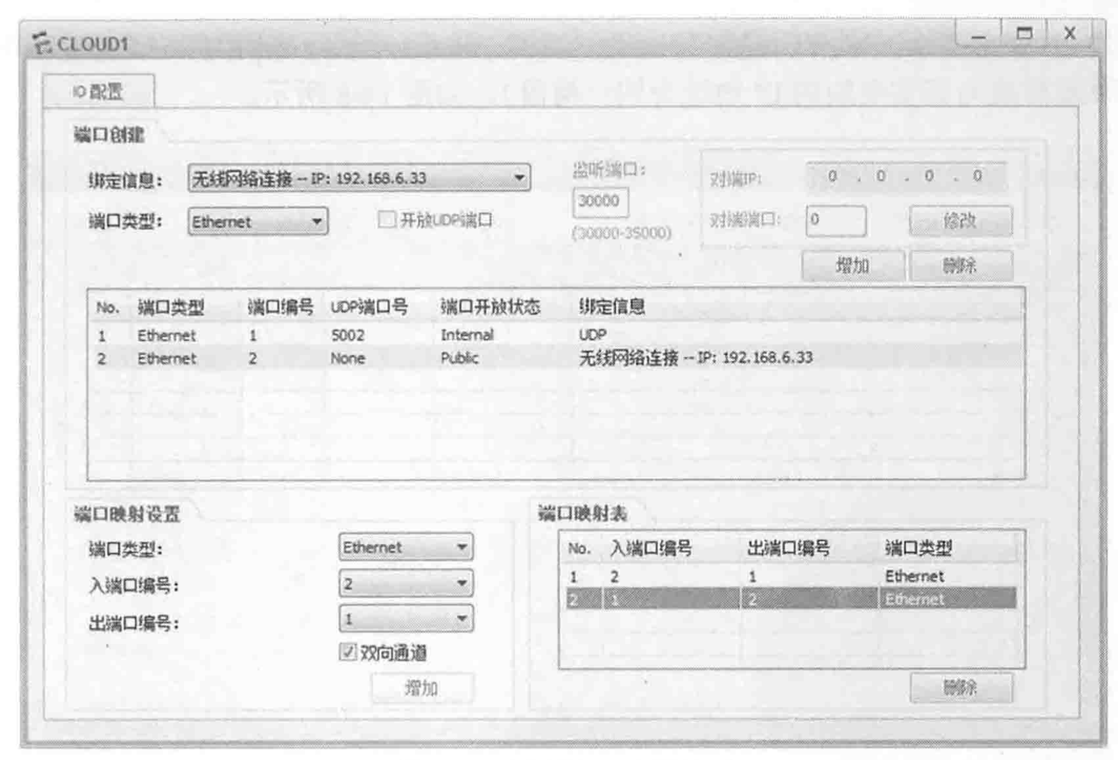


图 14-6　创建端口映射关系

端口的映射关系表说明了模拟器的设备与真实的 PC 之间的通信连接方式，指明了数据从哪个接口发送，从哪个接口接收。

2. 为模拟器添加设备，实现与真实设备的桥接

在 eNSP 模拟器中添加一台虚拟 PC，使用线缆连接到云设备的 E 0/0/1 接口，如图 14-7 所示，并启动设备。

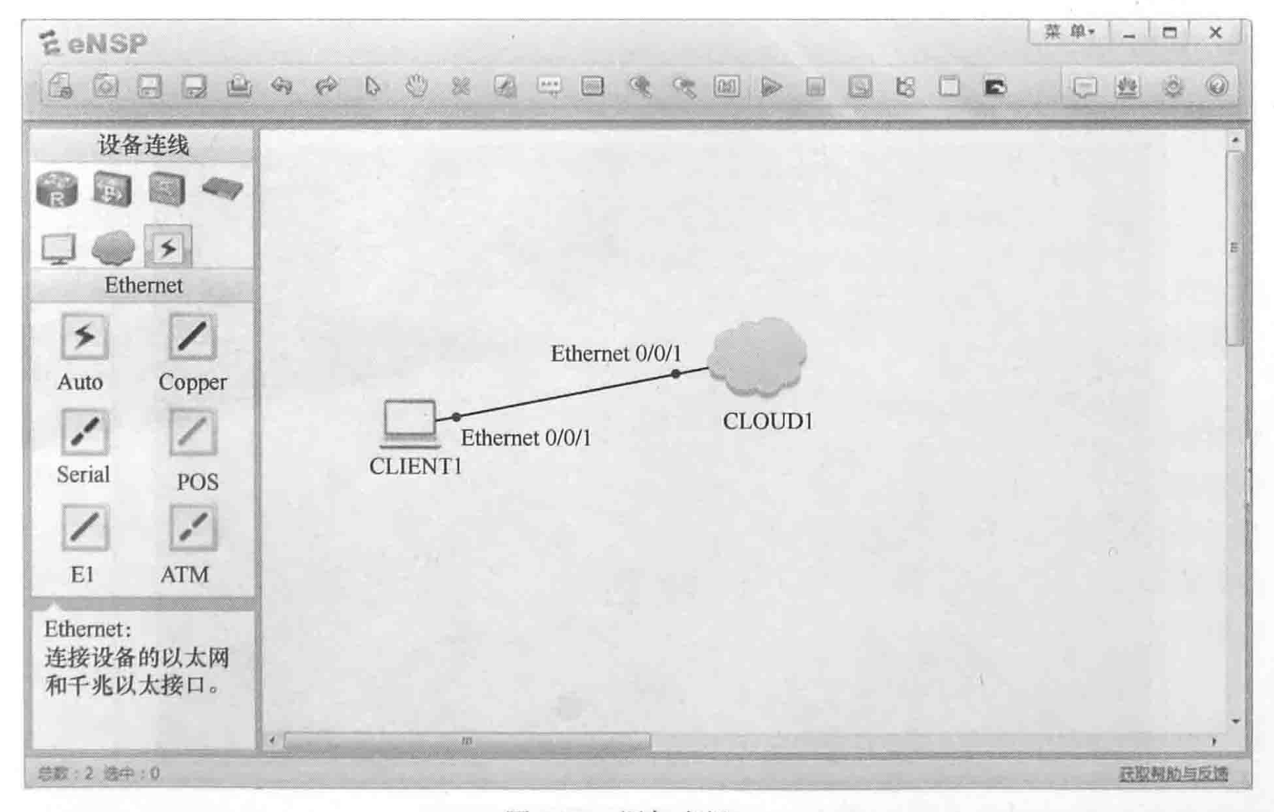


图 14-7　添加虚拟 PC

为 eNSP 模拟器中的 PC 设置 IP 地址为 192.168.6.1，子网掩码为 255.255.255.0（该地址要配置成与真实电脑的 IP 地址为同一网段），如图 14-8 所示。

图 14-8　PC 配置界面

3. 验证模拟器中的 PC 与真实 PC 之间的连通性

在真实 PC 上使用 **ping** 命令，验证与模拟器中虚拟 PC 间的连通性，观察到可以连通，如图 14-9 所示。



图 14-9　测试与虚拟 PC 间的连通性

在模拟器中使用 **ping** 命令，验证与真实 PC 间的连通性，观察到可以连通，如图 14-10 所示。



图 14-10 测试与真实 PC 间的连通性

至此，成功实现了真实 PC 通过 eNSP 中云设备的桥接功能与虚拟 PC 间的通信。

14.2 SNMP 基础配置

原理概述

随着网络规模的日益发展，现有的网络中，设备数量日益庞大。当这些设备发生故障时，由于设备无法主动上报故障，导致网络管理员无法及时感知、定位和排除故障，从而导致网络的维护效率降低，维护工作量大大增加。

为了解决这个问题，设备制造商已经在一些设备中提供了网络管理的功能，这样网管就可以远程查看设备的状态，同样，设备也能够在特定类型的事件发生时向网络管理工作站发出警告。SNMP（Simple Network Management Protocol，简单网络管理协议）就是规定网管站和设备之间如何传递管理信息的应用层协议。SNMP 定义了网管管理设备的几种操作，以及设备故障时能向网管主动发送告警。网络管理使用 SNMP 协议时存在网络管理站（NMS）、代理进程（Agent）和被管理设备 3 个角色。

- 网络管理站（NMS）：向被管理设备发送各种查询报文，以及接收被管理设备发送的告警；
- 代理进程（Agent）：是被管理设备上的一个代理进程，用于维护被管理设备的信

息数据并响应来自NMS的请求，把管理数据汇报给发送请求的NMS。Agent的作用为接收、解析来自网管站的查询报文；根据报文类型对管理变量进行Read或Write操作，并生成响应报文，返回给网管站；根据各协议模块对告警触发条件的定义，在达到触发条件后，如进入、退出系统视图或设备重新启动等，相应的模块通过Agent主动向网管站发送告警，报告所发生的事件；

■ 被管理设备：接受网管的管理，产生和主动上报告警。

实验内容

本实验模拟真实网络场景。在网络当中分别部署了两台管理站设备（NMS）和一台代理站设备（Agent），分别使用两台PC来模拟NMS；一台路由器来模拟Agent。本实验将介绍如何配置Agent设备、版本，配置管理站及用户权限，了解SNMP协议的作用及管理方法。被管理的设备可以是路由器、服务器、交换机、主机等设备，一般与Agent部署在同一网络。

实验目的

- 理解SNMP 应用场景
- 掌握配置SNMP Agent的方法
- 掌握配置SNMP 版本的方法
- 掌握配置管理站、用户权限的方法

实验拓扑

SNMP基础配置的拓扑如图14-11所示。

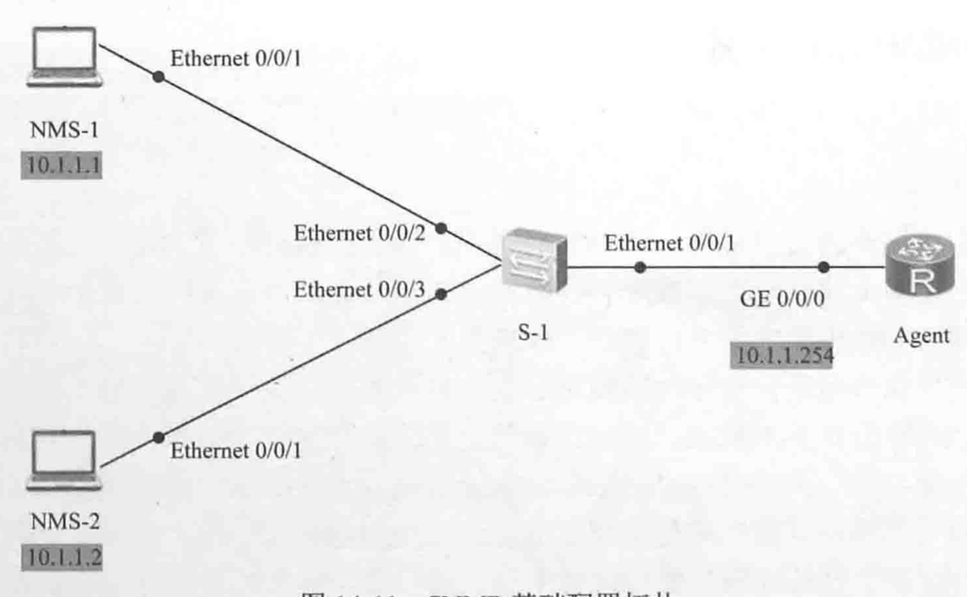


图14-11 SNMP基础配置拓扑

实验编址

实验编址见表14-1。

表 14-1　实验编址

设备	接口	IP 地址	子网掩码	默认网关
NMS-1	Ethernet 0/0/1	10.1.1.1	255.255.255.0	N/A
NMS-2	Ethernet 0/0/1	10.1.1.2	255.255.255.0	N/A
Agent（AR2220）	GE 0/0/0	10.1.1.254	255.255.255.0	N/A

实验步骤

1. 基本配置

将两台 PC 模拟成管理站，分别为 NMS-1、NMS-2，根据实验编址表配置设备名称及 IP 地址，并验证连通性。

```
[Agent]ping 10.1.1.1
  PING 10.1.1.1: 56  data bytes, press CTRL_C to break
    Reply from 10.1.1.1: bytes=56 Sequence=1 ttl=128 time=550 ms
    Reply from 10.1.1.1: bytes=56 Sequence=2 ttl=128 time=200 ms
    Reply from 10.1.1.1: bytes=56 Sequence=3 ttl=128 time=150 ms
    Reply from 10.1.1.1: bytes=56 Sequence=4 ttl=128 time=90 ms
    Reply from 10.1.1.1: bytes=56 Sequence=5 ttl=128 time=120 ms
  --- 10.1.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 90/222/550 ms
```

其余直连网段的连通性测试省略。

2. 开启 Agent 服务

将路由器模拟为代理站设备，需要为其启动 SNMP Agent 服务。

```
[Agent]snmp-agent
```

配置完成后使用 **display snmp-agent sys-info** 命令查看 SNMP 信息。

```
[Agent]display snmp-agent sys-info
   The contact person for this managed node:
           R&D Shenzhen, Huawei Technologies Co., Ltd.
   The physical location of this node:
           Shenzhen China
 SNMP version running in the system:
SNMPv1 SNMPv2c SNMPv3
```

显示当前默认情况下所运行的 SNMP 版本为 v1、v2c、v3。

3. 配置 SNMP 版本

SNMP 一共有 3 个版本，分别为 v1、v2c、v3，开启 SNMP 服务后默认同时兼容 3 个版本。当网络规模较小且安全性较高时，在规划时可配置设备使用 SNMPv1 版本与网管进行通信；当网络规模较大，安全性较高时，但运行的业务较为繁忙，在规划时配置设备使用 SNMPv2c 版本与网管进行通信；当网络规模较大且安全性较低时，在规划时配置设备使用 SNMPv3 版本与网管进行通信，并配置认证和加密功能保证安全性。

根据实际网络需求，使用 **snmp-agent sys-info** 命令配置 SNMP 版本，本例中使用 SNMPv3 版本。

```
[Agent]snmp-agent sys-info version v3
```

配置完后查看 Agent 信息。

```
[Agent]display snmp-agent sys-info version
    SNMP version running in the system:
            SNMPv3
```

显示当前运行的 SNMP 版本为 v3。

由于 SNMPv3 版本适用于大型网络规模，且可以配置认证加密，在实际工作环境下通常使用该版本。

4. 配置 NMS 管理权限

如果网络中不止一个管理站用户，可以根据业务需要，为不同管理站用户设置不同的访问权限。本实验中有两个管理站用户，现仅允许 NMS-2 可以管理设备。

配置基本 ACL，限制 NMS-2 管理设备、NMS-1 不允许管理设备。

```
[Agent]acl 2000
[Agent-acl-basic-2000]rule 5 permit source 10.1.1.2 0.0.0.255
[Agent-acl-basic-2000]rule 10 deny source 10.1.1.1 0.0.0.255
```

配置用户组为 group，用户名为 user，指定使用 ACL2000。

```
[Agent]snmp-agent usm-user v3 user group acl 2000
```

配置完成后，查看 SNMPv3 的用户信息。

```
[Agent]display snmp-agent usm-user
User name: user
    Engine ID: 800007DB03000000000000
Group name: group
    Authentication mode: No authentication mode, Privacy mode: No privacy mode
    Storage type: nonVolatile
    User status: active
Acl: 2000
    Total number is 1
```

可以观察到，配置已经生效。

5. 配置向 SNMP Agent 输出 Trap 信息

网络管理员需要查看被管理者产生的 Trap 信息，以便监控设备运行情况及定位故障信息。

配置 Agent 发送 Trap 消息。用于接收该 Trap 消息的网管名为 adminNMS2，目标地址为 10.1.1.2，且指定接收该消息使用 UDP 端口为 9991。Trap 消息的发送参数信息列表名称为 trapNMS2。

```
[Agent]snmp-agent target-host trap-hostname adminNMS2 address 10.1.1.2 udp-port 9991 trap-paramsname trapNMS2
```

开启设备的告警开关。只有将该开关打开以后，Agent 才会向网管站发送告警消息。

```
[Agent]snmp-agent trap enable
Info: All switches of SNMP trap/notification will be open. Continue? [Y/N]:y
```

设置告警消息的队列长度为 200（默认值为 100）。如果某个时间段 trap 报文消息很多，为防止丢包，可以设置增加消息队列长度以便减少丢包的情况发生。

```
[Agent]snmp-agent trap queue-size 200
```

设置报文消息的保存时间为 240 秒（默认值为 120）。该值是 Trap 报文消息的生存时间，如果超过该时间报文将会被丢弃，不再发送，也不再保存。

```
[Agent]snmp-agent trap life 240
```

为了便于维护，配置管理员的联系方式，电话为 400-822-9999，地址为中国深圳。

```
[Agent]snmp-agent sys-info contact call admin 400-822-9999
```

```
[Agent]snmp-agent sys-info location ShenZhen China
```

配置完后使用 **display snmp-agent sys-info** 命令查看相关的系统信息。

```
[Agent]display snmp-agent sys-info
    The contact person for this managed node:
    call admin 400-822-9999
    The physical location of this node:
    ShenZhen China
    SNMP version running in the system:
SNMPv3
```

查看 SNMP Agent 输出网管的信息。

```
[Agent]display snmp-agent target-host
    Traphost list:
    Target host name: adminNMS2
    Traphost address: 10.1.1.2
    Traphost portnumber: 9991
    Target host parameter: trapNMS2
    Total number is 1
    Parameter list trap target host:
    Total number is 0
```

配置完成后通过命令可以观察到 Trap 目标主机名为 adminNMS2，主机地址 10.1.1.2，主机端口为 9991，目标主机参数列表名为 trapNMS2。

14.3 GRE 协议基础配置

原理概述

GRE（Generic Routing Encapsulation，通用路由封装协议）提供了将一种协议的报文封装在另一种协议报文中的机制，使报文能够在异种网络（如 IPv4 网络）中传输，而异种报文传输的通道称为 Tunnel。

GRE 协议也可以作为 VPN 的第三层隧道（Tunnel）协议，为 VPN 数据提供透明传输通道。Tunnel 是一个虚拟的点对点的连接，可以看成仅支持点对点连接的虚拟接口，这个接口提供了一条通路，使封装的数据报能够在这个通路上传输，并在一个 Tunnel 的两端分别对数据报进行封装及解封装。

实验目的

- 理解 GRE 协议的使用场景
- 掌握配置 GRE 隧道的方法
- 掌握配置基于 GRE 接口的动态路由协议的方法

实验内容

本实验模拟企业网络场景。R1 为企业总部的网关设备，并且内部有一台服务器，R3 连接着企业分公司的网关设备，R2 为公网 ISP 设备。一般情况下，运营商只会维护自身的公网路由信息，而不会维护企业内部私网的路由信息，即运营商设备上的路由表中不会出现任何企业内部私网的路由条目。通过配置 GRE 实现公司总部和分部间私网路

由信息的透传及数据通信。

实验拓扑

GRE 协议基础配置的拓扑如图 14-12 所示。

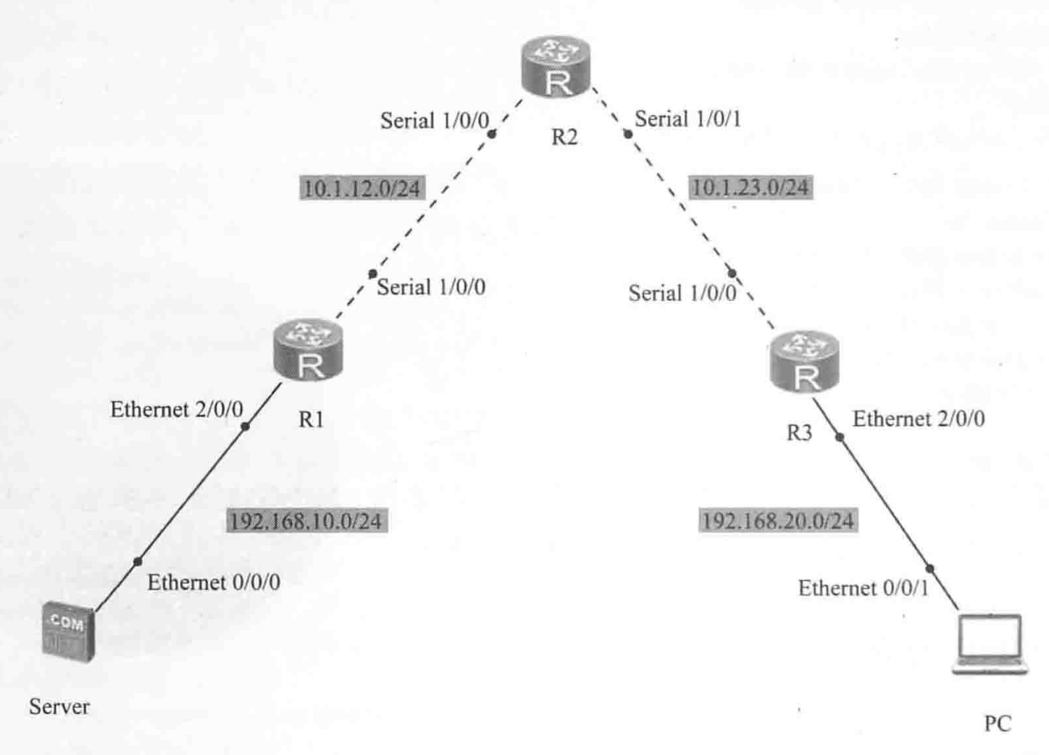


图 14-12 GRE 协议基础配置拓扑

实验编址

实验编址见表 14-2。

表 14-2 实验编址

设备	接口	IP 地址	子网掩码	默认网关
Server	Ethernet 0/0/0	192.168.10.10	255.255.255.0	192.168.10.1
R1（AR2220）	Ethernet 2/0/0	192.168.10.1	255.255.255.0	N/A
	Serial 1/0/0	10.1.12.1	255.255.255.0	N/A
R2（AR2220）	Serial 1/0/0	10.1.12.2	255.255.255.0	N/A
	Serial 1/0/1	10.1.23.2	255.255.255.0	N/A
R3（AR2220）	Serial 1/0/0	10.1.23.1	255.255.255.0	N/A
	Ethernet 2/0/0	192.168.20.1	255.255.255.0	N/A
PC	Ethernet 0/0/1	192.168.20.20	255.255.255.0	192.168.20.1

实验步骤

1. 基本配置

根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测各直连链路的连通性。

```
<R1>ping -c 1 192.168.10.10
  PING 192.168.10.10: 56    data bytes, press CTRL_C to break
    Reply from 192.168.10.10: bytes=56 Sequence=1 ttl=255 time=510 ms
  --- 192.168.10.10 ping statistics ---
    1 packet(s) transmitted
    1 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 510/510/510 ms
```

其余直连网段的连通性测试省略。

在 R1 和 R3 上分别配置访问公网路由器 R2 的默认路由。

```
[R1]ip route-static 0.0.0.0 0.0.0.0 10.1.12.2

[R3]ip route-static 0.0.0.0 0.0.0.0 10.1.23.2
```

配置完成后在 PC 上测试与总部服务器间的连通性。

```
PC>ping 192.168.10.10
Ping 192.168.10.10: 32 data bytes, Press Ctrl_C to break
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
……
```

可以观察到，跨越了互联网的两个私网网段之间默认是无法直接通信的。此时可以通过 GRE 协议来实现通信。

2. 配置 GRE Tunnel

在路由器 R1 和 R3 上配置 GRE Tunnel，使用 **interface tunnel** 命令创建隧道接口，指定隧道模式为 GRE。配置 R1 Tunnel 接口的源地址为其 S 1/0/0 接口 IP 地址，目的地址为 R3 的 S 1/0/0 接口 IP 地址；配置 R3 Tunnel 接口源地址为其 S 1/0/0 接口 IP 地址，目的地址为 R1 的 S 1/0/0 接口 IP 地址。还要使用 **ip address** 命令配置 Tunnel 接口的 IP 地址，注意要在同一网段。

```
[R1]interface tunnel 0/0/0
[R1-Tunnel0/0/0]tunnel-protocol gre
[R1-Tunnel0/0/0]source 10.1.12.1
[R1-Tunnel0/0/0]destination 10.1.23.1
[R1-Tunnel0/0/0]ip address 172.16.1.1 24

[R3]interface tunnel 0/0/0
[R3-Tunnel0/0/0]tunnel-protocol gre
[R3-Tunnel0/0/0]source 10.1.23.1
[R3-Tunnel0/0/0]destination 10.1.12.1
[R3-Tunnel0/0/0]ip address 172.16.1.2 24
```

配置完成后，在 R1 上测试本端隧道接口地址与目的端口隧道接口地址的连通性。

```
[R1]ping -a 172.16.1.1 172.16.1.2
  PING 172.16.1.2: 56    data bytes, press CTRL_C to break
    Reply from 172.16.1.2: bytes=56 Sequence=1 ttl=255 time=60 ms
    Reply from 172.16.1.2: bytes=56 Sequence=2 ttl=255 time=60 ms
    Reply from 172.16.1.2: bytes=56 Sequence=3 ttl=255 time=50 ms
    Reply from 172.16.1.2: bytes=56 Sequence=4 ttl=255 time=50 ms
    Reply from 172.16.1.2: bytes=56 Sequence=5 ttl=255 time=60 ms
  --- 172.16.1.2 ping statistics ---
```

```
        5 packet(s) transmitted
        5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 50/56/60 ms
```

可以观察到，通信正常。

在 R1 和 R3 上分别执行 **display interface tunnel** 命令查看隧道接口状态。

```
[R1]display interface Tunnel 0/0/0
Tunnel0/0/0 current state : UP
Line protocol current state : UP
Last line protocol up time : 2013-05-29 15:36:53 UTC-08:00
Description:
Route Port,The Maximum Transmit Unit is 1500
Internet Address is 172.16.1.1/24
Encapsulation is TUNNEL, loopback not set
Tunnel source 10.1.12.1 (Serial1/0/0), destination 10.1.23.1
Tunnel protocol/transport GRE/IP, key disabled
keepalive disabled
……

[R3]display interface Tunnel 0/0/0
Tunnel0/0/0 current state : UP
Line protocol current state : UP
Last line protocol up time : 2013-05-29 16:06:09 UTC-08:00
Description:
Route Port,The Maximum Transmit Unit is 1500
Internet Address is 172.16.1.2/24
Encapsulation is TUNNEL, loopback not set
Tunnel source 10.1.23.1 (Serial1/0/0), destination 10.1.12.1
Tunnel protocol/transport GRE/IP, key disabled
keepalive disabled
……
```

可以观察到，当前隧道接口的物理层状态为正常启动状态，链路层协议状态为正常运行状态，隧道封装协议为 GRE 协议，Tunnel 的 IP 地址及所配置的隧道源和目的地址分别为 R1 和 R3 的 S1/0/0 接口地址。

在 R1 和 R2 上执行 **display ip routing-table** 命令查看路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 10        Routes : 10
Destination/Mask    Proto    Pre  Cost      Flags NextHop         Interface
      0.0.0.0/0     Static   60   0           RD   10.1.12.2       Serial1/0/0
    10.1.12.0/24    Direct   0    0           D    10.1.12.1       Serial1/0/0
    10.1.12.1/32    Direct   0    0           D    127.0.0.1       Serial1/0/0
    10.1.12.2/32    Direct   0    0           D    10.1.12.2       Serial1/0/0
    127.0.0.0/8     Direct   0    0           D    127.0.0.1       InLoopBack0
    127.0.0.1/32    Direct   0    0           D    127.0.0.1       InLoopBack0
   172.16.1.0/24    Direct   0    0           D    172.16.1.1      Tunnel0/0/0
   172.16.1.1/32    Direct   0    0           D    127.0.0.1       Tunnel0/0/0
  192.168.10.0/24   Direct   0    0           D    192.168.10.1    Ethernet2/0/0
  192.168.10.1/32   Direct   0    0           D    127.0.0.1       Ethernet2/0/0
```

```
[R3]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 10        Routes : 10
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
        0.0.0.0/0   Static  60   0           RD   10.1.23.2       Serial1/0/0
      10.1.23.0/24  Direct  0    0           D    10.1.23.1       Serial1/0/0
      10.1.23.1/32  Direct  0    0           D    127.0.0.1       Serial1/0/0
      10.1.23.2/32  Direct  0    0           D    10.1.23.2       Serial1/0/0
      127.0.0.0/8   Direct  0    0           D    127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D    127.0.0.1       InLoopBack0
      172.16.1.0/24 Direct  0    0           D    172.16.1.2      Tunnel0/0/0
      172.16.1.2/32 Direct  0    0           D    127.0.0.1       Tunnel0/0/0
    192.168.20.0/24 Direct  0    0           D    192.168.20.1    Ethernet2/0/0
    192.168.20.1/32 Direct  0    0           D    127.0.0.1       Ethernet2/0/0
```

可以观察到，R1 和 R3 的路由表中已经有所配置隧道接口的路由条目。即 R1 和 R3 间已经形成了类似点到点直连的逻辑链路，但没有互相接收到对方的私网路由信息。

3. 配置基于 GRE 接口的动态路由协议

经过上面的步骤，R1 和 R3 之间的 GRE 隧道已经建立，测试分部 PC 与总部服务器间的连通性。

```
PC>ping 192.168.10.10
Ping 192.168.10.10: 32 data bytes, Press Ctrl_C to break
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!
……
```

仍然无法正常通信。原因是目前还没有配置基于 GRE 隧道接口的路由协议。GRE 协议支持组播数据的传输，因此可以支持一些动态路由协议的运行，动态路由协议通过 GRE 协议形成的逻辑隧道在 R1 和 R3 间传递路由信息。

在 R1 和 R3 上配置 RIPv2 协议，通告相应的私网网段和 Tunnel 接口所在网络。

```
[R1]rip 1
[R1-rip-1]version 2
[R1-rip-1]network 192.168.10.0
[R1-rip-1]network 172.16.0.0

[R3]rip 1
[R3-rip-1]version 2
[R3-rip-1]network 192.168.20.0
[R3-rip-1]network 172.16.0.0
```

分别在 R1 与 R3 上查看 RIP 邻居。

```
[R1]display rip 1 neighbor
------------------------------------------------------------------------------
 IP Address      Interface                    Type   Last-Heard-Time
------------------------------------------------------------------------------
172.16.1.2      Tunnel0/0/0                  RIP     0:0:7
 Number of RIP routes   : 1

[R3]display rip 1 neighbor
```

```
----------------------------------------------------------------------
 IP Address        Interface                 Type    Last-Heard-Time
----------------------------------------------------------------------
172.16.1.1         Tunnel0/0/0               RIP     0:0:12
 Number of RIP routes    : 1
```

可以观察到，此时双方都已经通过隧道接口建立了 RIP 邻居关系。

查看路由器 R1 和 R3 路由表。

```
[R1]display ip routing-table
Route Flags: R - relay, D - download to fib
----------------------------------------------------------------------
Routing Tables: Public
         Destinations : 11        Routes : 11
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
        0.0.0.0/0   Static  60   0           RD   10.1.12.2       Serial1/0/0
      10.1.12.0/24  Direct  0    0           D    10.1.12.1       Serial1/0/0
      10.1.12.1/32  Direct  0    0           D    127.0.0.1       Serial1/0/0
      10.1.12.2/32  Direct  0    0           D    10.1.12.2       Serial1/0/0
      127.0.0.0/8   Direct  0    0           D    127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D    127.0.0.1       InLoopBack0
     172.16.1.0/24  Direct  0    0           D    172.16.1.1      Tunnel0/0/0
     172.16.1.1/32  Direct  0    0           D    127.0.0.1       Tunnel0/0/0
   192.168.10.0/24  Direct  0    0           D    192.168.10.1    Ethernet2/0/0
   192.168.10.1/32  Direct  0    0           D    127.0.0.1       Ethernet2/0/0
192.168.20.0/24     RIP     100  1           D    172.16.1.2      Tunnel0/0/0

[R3]display ip routing-table
Route Flags: R - relay, D - download to fib
----------------------------------------------------------------------
Routing Tables: Public
         Destinations : 11        Routes : 11
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
        0.0.0.0/0   Static  60   0           RD   10.1.23.2       Serial1/0/0
      10.1.23.0/24  Direct  0    0           D    10.1.23.1       Serial1/0/0
      10.1.23.1/32  Direct  0    0           D    127.0.0.1       Serial1/0/0
      10.1.23.2/32  Direct  0    0           D    10.1.23.2       Serial1/0/0
      127.0.0.0/8   Direct  0    0           D    127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D    127.0.0.1       InLoopBack0
     172.16.1.0/24  Direct  0    0           D    172.16.1.2      Tunnel0/0/0
     172.16.1.2/32  Direct  0    0           D    127.0.0.1       Tunnel0/0/0
   192.168.10.0/24  RIP     100  1           D    172.16.1.1      Tunnel0/0/0
   192.168.20.0/24  Direct  0    0           D    192.168.20.1    Ethernet2/0/0
192.168.20.1/32     Direct  0    0           D    127.0.0.1       Ethernet2/0/0
```

可以观察到，双方都能接收到各自内部私有网络发送过来的路由更新。

在分公司 PC 上测试与总公司 Server 间的连通性。

```
PC>ping 192.168.10.10
Ping 192.168.10.10: 32 data bytes, Press Ctrl_C to break
From 192.168.10.10: bytes=32 seq=1 ttl=126 time=47 ms
From 192.168.10.10: bytes=32 seq=2 ttl=126 time=46 ms
From 192.168.10.10: bytes=32 seq=3 ttl=126 time=47 ms
From 192.168.10.10: bytes=32 seq=4 ttl=126 time=62 ms
From 192.168.10.10: bytes=32 seq=5 ttl=126 time=47 ms
--- 192.168.10.10 ping statistics ---
  5 packet(s) transmitted
```

```
  5 packet(s) received
0.00% packet loss
  round-trip min/avg/max = 46/49/62 ms
```

可以观察到，通信正常。查看路由器 R2 的路由表。

```
[R2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8          Routes : 8
Destination/Mask    Proto     Pre   Cost      Flags NextHop         Interface
      10.1.12.0/24  Direct    0     0           D   10.1.12.2       Serial1/0/0
      10.1.12.1/32  Direct    0     0           D   10.1.12.1       Serial1/0/0
      10.1.12.2/32  Direct    0     0           D   127.0.0.1       Serial1/0/0
      10.1.23.0/24  Direct    0     0           D   10.1.23.2       Serial1/0/1
      10.1.23.1/32  Direct    0     0           D   10.1.23.1       Serial1/0/1
      10.1.23.2/32  Direct    0     0           D   127.0.0.1       Serial1/0/1
      127.0.0.0/8   Direct    0     0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct    0     0           D   127.0.0.1       InLoopBack0
```

可以观察到 R2 上没有任何 R1 和 R3 的各自私网网段的路由信息。说明通过 GRE 协议建立起来的隧道，能够跨公网传递各个内部私有网络的路由信息，实现了两个私有网络间的跨公网通信。

思考

GRE 是一种三层隧道协议，可以形成逻辑的点到点直连隧道，支持组播数据的传输，但是它的安全性能较差，不能实现隧道中所传输数据的加密。那么 GRE 应采用何种方式实现组播数据跨互联网的加密传输？

14.4 配置 NAT

原理概述

早在 20 世纪 90 年代初，有关 RFC 文档就提出了 IP 地址耗尽的可能性。IPv6 技术的提出虽然可以从根本上解决地址短缺的问题，但是也无法立刻替换现有成熟且广泛应用的 IPv4 网络。既然不能立即过渡到 IPv6 网络，那么必须使用一些技术手段来延长 IPv4 的寿命，其中广泛使用的技术之一就是网络地址转换（Network Address Translation，NAT）。

NAT 是将 IP 数据报文报头中的 IP 地址转换为另一个 IP 地址的过程，主要用于实现内部网络（私有 IP 地址）访问外部网络（公有 IP 地址）的功能。NAT 有 3 种类型：静态 NAT、动态地址 NAT 以及网络地址端口转换 NAPT。

NAT 转换设备（实现 NAT 功能的网络设备）维护着地址转换表，所有经过 NAT 转换设备并且需要进行地址转换的报文，都会通过该表做相应转换。NAT 转换设备处于内部网络和外部网络的连接处，常见的有路由器、防火墙等。

实验目的

- 理解 NAT 的应用场景
- 掌握静态 NAT 的配置
- 掌握 NAT Outbound 的配置
- 掌握 NAT Easy-IP 的配置
- 掌握 NAT Server 的配置

实验内容

本实验模拟企业网络场景。R1 是公司的出口网关路由器，公司内员工和服务器都通过交换机 S1 或 S2 连接到 R1 上，R2 模拟外网设备与 R1 直连。由于公司内网都使用私网 IP 地址，为了实现公司内部分员工可以访问外网，服务器可以供外网用户访问，网络管理员需要在路由器 R1 上配置 NAT：使用静态 NAT 和 NAT Outbound 技术使部分员工可以访问外网，使用 NAT Server 技术使服务器可以供外网用户访问。

实验拓扑

配置 NAT 的拓扑如图 14-13 所示。

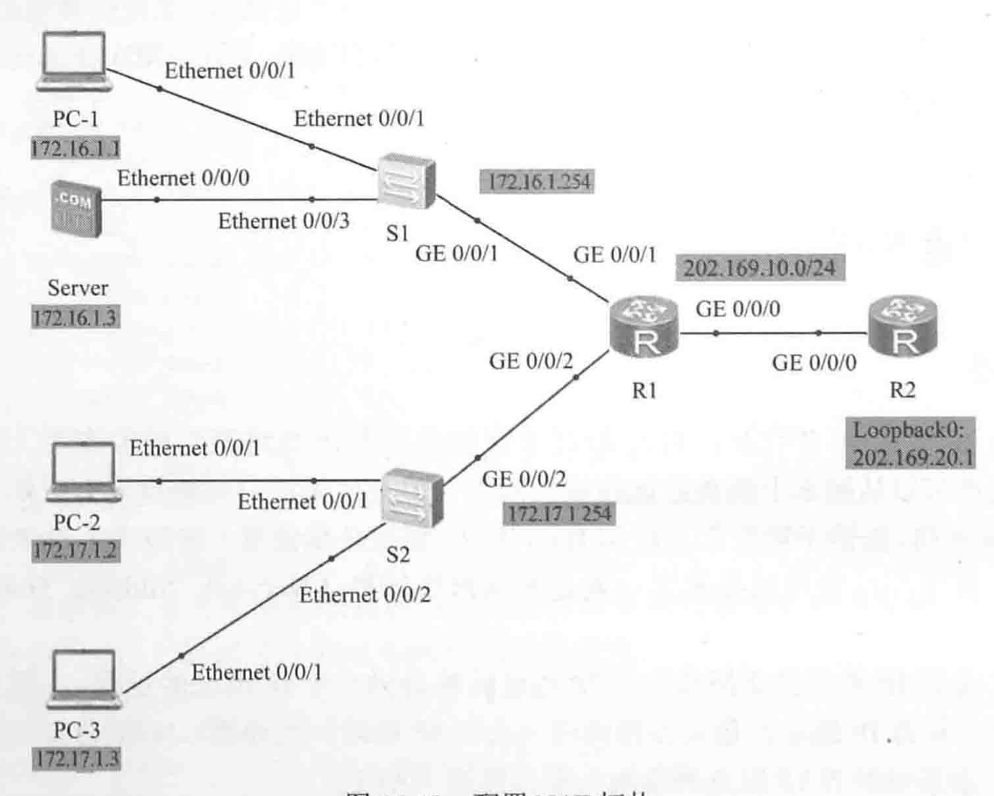


图 14-13　配置 NAT 拓扑

实验编址

实验编址见表 14-3。

表 14-3 实验编址

设备	接口	IP 地址	子网掩码	默认网关
R1（AR2200）	GE 0/0/0	202.169.10.1	255.255.255.0	N/A
	GE 0/0/1	172.16.1.254	255.255.255.0	N/A
	GE 0/0/2	172.17.1.254	255.255.255.0	N/A
R2（AR2200）	GE 0/0/0	202.169.10.2	255.255.255.0	N/A
	Loopback 0	202.169.20.1	255.255.255.0	N/A
PC-1	Ethernet 0/0/1	172.16.1.1	255.255.255.0	172.16.1.254
PC-2	Ethernet 0/0/1	172.17.1.2	255.255.255.0	172.17.1.254
PC-3	Ethernet 0/0/1	172.17.1.3	255.255.255.0	172.17.1.254
Server	Ethernet 0/0/0	172.16.1.3	255.255.255.0	172.16.1.254

实验步骤

1. 基本配置

根据实验编址表进行相应的基本配置，并使用 **ping** 命令检测各直连链路的连通性。

```
<R1>ping -c 1 202.169.10.2
  PING 202.169.10.2: 56  data bytes, press CTRL_C to break
    Reply from 202.169.10.2: bytes=56 Sequence=1 ttl=255 time=140 ms
  --- 202.169.10.2 ping statistics ---
    1 packet(s) transmitted
    1 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 140/140/140 ms
```

其余直连网段的连通性测试省略。

2. 配置静态 NAT

公司在网关路由器 R1 上配置访问外网的默认路由。

```
[R1]ip route-static 0.0.0.0 0.0.0.0 202.169.10.2
```

由于内网使用的都是私有 IP 地址，员工无法直接访问公网。现需要在网关路由器 R1 上配置 NAT 地址转换，将私网地址转换为公网地址。

PC-1 为公司客户经理使用的终端，不仅需要自身能访问外网，还需要外网用户也能够直接访问他，因此网络管理员分配了一个公网 IP 地址 202.169.10.5 给 PC-1 做静态 NAT 地址转换。在 R1 的 GE 0/0/0 接口下使用 **nat static** 命令配置内部地址到外部地址的一对一转换。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]nat static global 202.169.10.5 inside 172.16.1.1
```

配置完成后，在 R1 上查看 NAT 静态配置信息，并在 PC-1 上使用 **ping** 命令测试与外网的连通性。

```
<R1>display nat static
  Static Nat Information:
  Interface   : GigabitEthernet0/0/0
    Global IP/Port       : 202.169.10.5/----
    Inside IP/Port       : 172.16.1.1/----
    Protocol : ----
    ……

PC>ping 202.169.20.1
Ping 202.169.20.1: 32 data bytes, Press Ctrl C to break
```

```
From 202.169.20.1: bytes=32 seq=1 ttl=254 time=422 ms
From 202.169.20.1: bytes=32 seq=2 ttl=254 time=156 ms
From 202.169.20.1: bytes=32 seq=3 ttl=254 time=203 ms
From 202.169.20.1: bytes=32 seq=4 ttl=254 time=47 ms
From 202.169.20.1: bytes=32 seq=5 ttl=254 time=63 ms
--- 202.169.20.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 47/178/422 ms
```

可以观察到，PC-1 通过静态 NAT 地址转换已经可以成功访问外网。在路由器 R1 的 GE 0/0/0 接口上抓包查看 NAT 地址转换是否成功，结果如图 14-14 所示。

No.	Time	Source	Destination	Protocol	Info
1	0.000000	202.169.10.5	202.169.20.1	ICMP	Echo (ping) request (id=0x9bbb, seq(be/le)=1/256, ttl=127)
2	0.016000	202.169.20.1	202.169.10.5	ICMP	Echo (ping) reply (id=0x9bbb, seq(be/le)=1/256, ttl=255)
3	1.030000	202.169.10.5	202.169.20.1	ICMP	Echo (ping) request (id=0x9cbb, seq(be/le)=2/512, ttl=127)
4	1.045000	202.169.20.1	202.169.10.5	ICMP	Echo (ping) reply (id=0x9cbb, seq(be/le)=2/512, ttl=255)
5	2.075000	202.169.10.5	202.169.20.1	ICMP	Echo (ping) request (id=0x9dbb, seq(be/le)=3/768, ttl=127)
6	2.075000	202.169.20.1	202.169.10.5	ICMP	Echo (ping) reply (id=0x9dbb, seq(be/le)=3/768, ttl=255)
7	3.120000	202.169.10.5	202.169.20.1	ICMP	Echo (ping) request (id=0x9ebb, seq(be/le)=4/1024, ttl=127)
8	3.120000	202.169.20.1	202.169.10.5	ICMP	Echo (ping) reply (id=0x9ebb, seq(be/le)=4/1024, ttl=255)
9	4.150000	202.169.10.5	202.169.20.1	ICMP	Echo (ping) request (id=0x9fbb, seq(be/le)=5/1280, ttl=127)
10	4.150000	202.169.20.1	202.169.10.5	ICMP	Echo (ping) reply (id=0x9fbb, seq(be/le)=5/1280, ttl=255)

```
⊞ Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)
⊞ Ethernet II, Src: HuaweiTe_b5:1a:a8 (00:e0:fc:b5:1a:a8), Dst: HuaweiTe_51:3a:6d (00:e0:fc:51:3a:6d)
⊟ Internet Protocol, Src: 202.169.10.5 (202.169.10.5), Dst: 202.169.20.1 (202.169.20.1)
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    Total Length: 60
    Identification: 0xbb9b (48027)
  ⊞ Flags: 0x02 (Don't Fragment)
    Fragment offset: 0
    Time to live: 127
    Protocol: ICMP (1)
  ⊞ Header checksum: 0x8ccc [correct]
    Source: 202.169.10.5 (202.169.10.5)
    Destination: 202.169.20.1 (202.169.20.1)
⊞ Internet Control Message Protocol
```

图 14-14　抓包观察

可以观察到 R1 已经成功把来自 PC-1 的 ICMP 报文的源地址 172.16.1.1 转换成公网地址 202.169.10.5。在 R2 上使用环回口 Loopback 0 模拟外网用户访问 PC-1，并在 PC-1 的 E 0/0/1 接口上抓包观察，如图 14-15 所示。

```
<R2>ping -a 202.169.20.1 202.169.10.5
  PING 202.169.10.5: 56  data bytes, press CTRL_C to break
    Reply from 202.169.10.5: bytes=56 Sequence=1 ttl=127 time=260 ms
    Reply from 202.169.10.5: bytes=56 Sequence=2 ttl=127 time=140 ms
    Reply from 202.169.10.5: bytes=56 Sequence=3 ttl=127 time=110 ms
    Reply from 202.169.10.5: bytes=56 Sequence=4 ttl=127 time=130 ms
    Reply from 202.169.10.5: bytes=56 Sequence=5 ttl=127 time=40 ms
  --- 202.169.10.5 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 40/136/260 ms
```

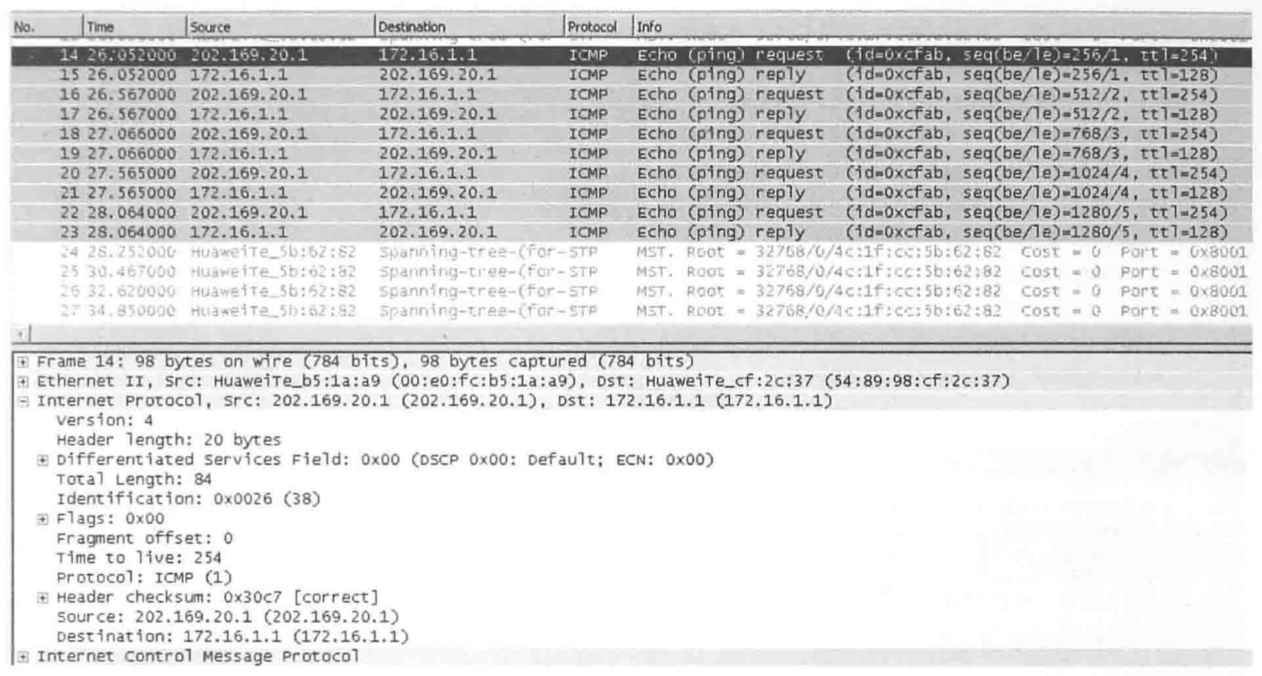


图 14-15　抓包观察

可以观察到由于 PC-1 的私网地址被转换为唯一的公网地址，外网用户也能主动访问 PC-1，且数据包在经过 R1 进入内网的时候，R1 把目的 IP 转换为与公网地址 202.169.10.5 对应的私网地址 172.16.1.1 发给 PC-1。

3. 配置 NAT Outbound

公司内市场部的员工都需要能够访问外网。市场部使用私网 IP 地址 172.17.1.0/24 网段，网络管理员使用公网地址池 202.169.10.50~202.169.10.60 为市场部员工做 NAT 转换。

在 R1 上使用 **nat addres-group** 命令配置 NAT 地址池，设置起始和结束地址分别为 202.169.10.50 和 202.169.10.60。

```
[R1]nat address-group 1 202.169.10.50 202.169.10.60
```

创建基本 ACL 2000，匹配 20.1.1.0，掩码为 24 位的地址段。

```
[R1]acl 2001
[R1-acl-basic-2001]rule 5 permit source 172.17.1.0 0.0.0.255
```

在 GE 0/0/0 接口下使用 **nat outbound** 命令将 ACL 2001 与地址池相关联，使得 ACL 中规定的地址可以使用地址池进行地址转换。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]nat outbound 2001 address-group 1 no-pat
```

配置完成后，在 R1 上查看 NAT Outbound 信息。

```
[R1]display nat outbound
 NAT Outbound Information:
 --------------------------------------------------------------------------
 Interface                Acl      Address-group/IP/Interface      Type

 GigabitEthernet0/0/0     2001                               1     no-pat
 --------------------------------------------------------------------------

  Total : 1
```

可以观察到 R1 上的 NAT Outbound 配置信息。使用 PC-2 测试与外网的连通性，并在 R1 的接口 GE 0/0/0 上抓包观察地址转换情况，如图 14-16 所示。

```
PC>ping 202.169.20.1
```

```
Ping 202.169.20.1: 32 data bytes, Press Ctrl_C to break
From 202.169.20.1: bytes=32 seq=1 ttl=254 time=219 ms
From 202.169.20.1: bytes=32 seq=2 ttl=254 time=141 ms
From 202.169.20.1: bytes=32 seq=3 ttl=254 time=94 ms
From 202.169.20.1: bytes=32 seq=4 ttl=254 time=47 ms
From 202.169.20.1: bytes=32 seq=5 ttl=254 time=94 ms
--- 202.169.20.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 47/119/219 ms
```

No.	Time	Source	Destination	Protocol	Info
1	0.000000	202.169.10.50	202.169.20.1	ICMP	Echo (ping) request (id=0x98db, seq(be/le)
2	0.125000	HuaweiTe_03:e5:0a	Broadcast	ARP	Who has 202.169.10.50? Tell 202.169.10.2
3	0.125000	HuaweiTe_03:64:1e	HuaweiTe_03:e5:0a	ARP	202.169.10.50 is at 00:e0:fc:03:64:1e
4	0.156000	202.169.20.1	202.169.10.50	ICMP	Echo (ping) reply (id=0x98db, seq(be/le)
5	1.187000	202.169.10.51	202.169.20.1	ICMP	Echo (ping) request (id=0x99db, seq(be/le)
6	1.265000	HuaweiTe_03:e5:0a	Broadcast	ARP	Who has 202.169.10.51? Tell 202.169.10.2
7	1.297000	HuaweiTe_03:64:1e	HuaweiTe_03:e5:0a	ARP	202.169.10.51 is at 00:e0:fc:03:64:1e

```
⊞ Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)
⊞ Ethernet II, Src: HuaweiTe_03:64:1e (00:e0:fc:03:64:1e), Dst: HuaweiTe_03:e5:0a (00:e0:fc:03:e5:0a)
⊟ Internet Protocol, Src: 202.169.10.50 (202.169.10.50), Dst: 202.169.20.1 (202.169.20.1)
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    Total Length: 60
    Identification: 0xdb98 (56216)
  ⊞ Flags: 0x02 (Don't Fragment)
    Fragment offset: 0
    Time to live: 127
    Protocol: ICMP (1)
  ⊞ Header checksum: 0x6ca2 [correct]
    Source: 202.169.10.50 (202.169.10.50)
    Destination: 202.169.20.1 (202.169.20.1)
⊞ Internet Control Message Protocol
```

图 14-16 抓包观察

可以观察到 PC-2 可以成功访问外网，且通过抓包分析，来自 PC-2 的 ICMP 数据包在 R1 的 GE 0/0/0 接口上源地址 172.17.1.2 被替换为地址池中第一个地址 202.169.10.50。

4. 配置 NAT Easy - IP

由于公司发展人员扩招，若继续使用多对多的 NAT 转换方式，就必须增加公网地址池的地址数。为了节约公网地址，网络管理员使用多对一的 Easy-IP 转换方式实现市场部员工访问外网的需求。

Easy-IP 是 NAPT 的一种方式，直接借用路由器出接口 IP 地址作为公网地址，将不同的内部地址映射到同一公有地址的不同端口号上，实现多对一地址转换。网络管理员配置路由器 R1 的 GE 0/0/0 接口为 Easy-IP 接口。

在 R1 的 GE 0/0/0 接口上删除 NAT Outbound 配置，并使用 **nat outbound** 命令配置 Easy-IP 特性，直接使用接口 IP 地址作为 NAT 转换后的地址。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]undo nat outbound 2001 address-group 1 no-pat
[R1-GigabitEthernet0/0/0]nat outbound 2001
```

配置完成后，在 PC-2 和 PC-3 上使用 UDP 发包工具发送 UDP 数据包到公网地址 202.169.20.1，配置好目的 IP 和 UDP 源、目的端口号后，输入字符串数据后单击“发送”按钮，如图 14-17、图 14-18 所示。

PC-2

基础配置　命令行　组播　UDP发包工具

地址

IPv4　IPv6

目的MAC地址: FF-FF-FF-FF-FF-FF　源MAC地址: 54-89-98-CF-DC-04

目的IP地址: 202 . 169 . 20 . 1　源IP地址: 172 . 17 . 1 . 2

目的端口号: 10 (0~65535)　源端口号: 10 (0~65535)

VLAN

VLAN　Vlan ID:　优先级:　(0~7)

数据包信息

数据包长度: 56 (28~65535)　MTU: 1500 (28~65535)

数据

输入十六进制字符串数据:

123456

周期发送　时间间隔: 1000 ms　发包个数: 1　发送　停止

图 14-17　PC-2 配置界面

PC-3

基础配置　命令行　组播　UDP发包工具

地址

IPv4　IPv6

目的MAC地址: FF-FF-FF-FF-FF-FF　源MAC地址: 54-89-98-CF-CC-4D

目的IP地址: 202 . 169 . 20 . 1　源IP地址: 172 . 17 . 1 . 3

目的端口号: 10 (0~65535)　源端口号: 10 (0~65535)

VLAN

VLAN　Vlan ID:　优先级:　(0~7)

数据包信息

数据包长度: 56 (28~65535)　MTU: 1500 (28~65535)

数据

输入十六进制字符串数据:

234567

周期发送　时间间隔: 1000 ms　发包个数: 1　发送　停止

图 14-18　PC-3 配置界面

在 PC-2 和 PC-3 发送 UDP 数据包后，在 R1 上查看 NAT Session 的详细信息。

```
<R1>display nat session protocol udp verbose
  NAT Session Table Information:
     Protocol            : UDP(17)
     SrcAddr   Port Vpn : 172.17.1.2          2560
     DestAddr Port Vpn : 202.169.20.1      2560
     Time To Live         : 120 s
     NAT-Info
     New SrcAddr         : 202.169.10.1
     New SrcPort         : 10255
     New DestAddr        : ----
     New DestPort        : ----
     Protocol            : UDP(17)
     SrcAddr   Port Vpn : 172.17.1.3          2560
     DestAddr Port Vpn : 202.169.20.1      2560
     Time To Live         : 120 s
     NAT-Info
     New SrcAddr         : 202.169.10.1
     New SrcPort         : 10256
     New DestAddr        : ----
     New DestPort        : ----
     Total : 2
```

可以观察到，源地址为 172.17.1.2 的 UDP 数据包被新源地址 202.169.10.1 和新源端口号 10255 替换，源地址为 172.17.1.3 的 UDP 数据包被新源地址 202.169.10.1 和新源端口号 10256 替换。R1 借用自身 GE 0/0/0 接口的公网 IP 地址为所有私网地址做 NAT 转换，使用不同的端口号区分不同私网数据。此方式不需要创建地址池，大大节省了地址空间。

5. 配置 NAT Server

公司内 Server 提供 FTP 服务供外网用户访问，配置 NAT Server 并使用公网 IP 地址 202.169.10.6 对外公布服务器地址，然后开启 NAT ALG 功能。因为对于封装在 IP 数据报文中的应用层协议报文，正常的 NAT 转换会导致错误，在开启某应用协议的 NAT ALG 功能后，该应用协议报文可以正常进行 NAT 转换，否则该应用协议不能正常工作。

在 R1 的 GE 0/0/0 接口上，使用 **nat server** 命令定义内部服务器的映射表，指定服务器通信协议类型为 TCP，配置服务器使用的公网 IP 地址为 202.169.10.6，服务器内网地址为 172.16.1.3，指定端口号为 21，该常用端口号可以直接使用关键字“ftp”代替。

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]nat server protocol tcp global 202.169.10.6 ftp inside 172.16.1.3 ftp
[R1-GigabitEthernet0/0/0]quit
[R1]nat alg ftp enable
```

配置完成后，在 R1 上查看 NAT Server 信息。

```
<R1>display nat server
  Nat Server Information:
  Interface  : GigabitEthernet0/0/0
    Global IP/Port       : 202.169.10.6/21(ftp)
    Inside IP/Port       : 172.16.1.3/21(ftp)
    Protocol : 6(tcp)
    VPN instance-name   : ----
    Acl number           : ----
    Description : ----
    Total :     1
```

可以观察到，配置已经生效，并开启服务器的 FTP 功能，如图 14-19 所示。

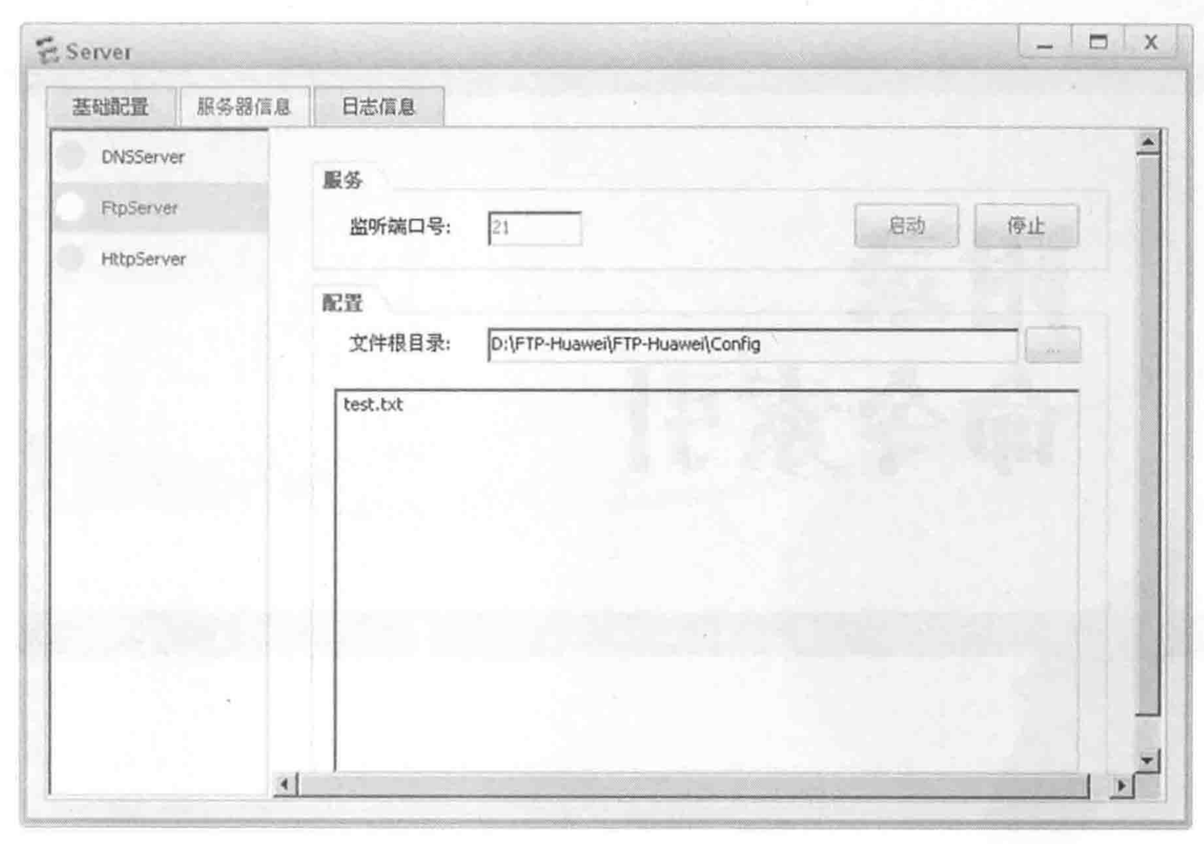


图 14-19　Server 配置界面

设置完服务器后，在 R2 上模拟公网用户访问该私网服务器。

```
<R2>ftp 202.169.10.6
Trying 202.169.10.6 ...
Press CTRL+K to abort
Connected to 202.169.10.6.
220 FtpServerTry FtpD for free
User(202.169.10.6:(none)):huawei
331 Password required for huawei .
Enter password:
230 User huawei logged in , proceed

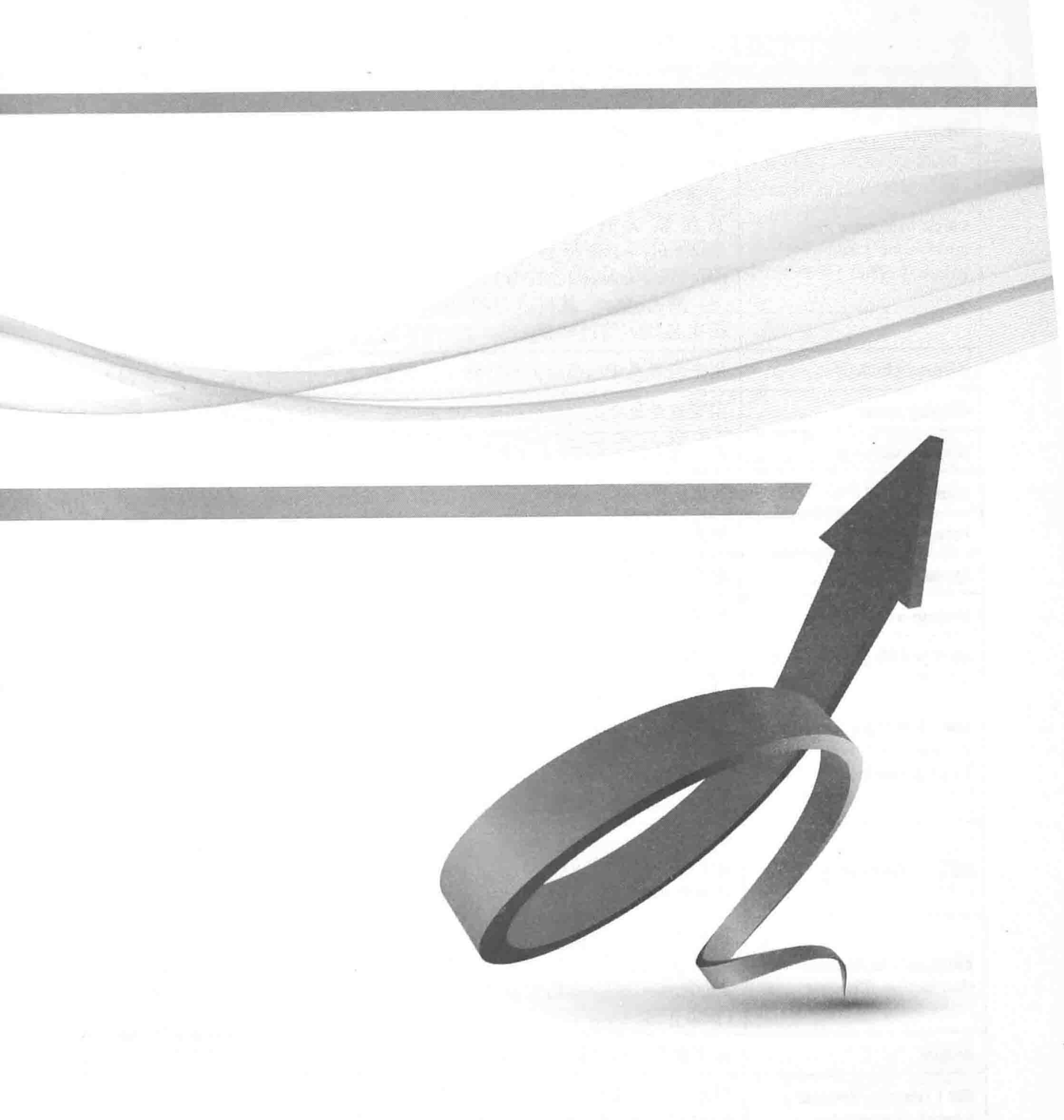
[R2-ftp]ls
200 Port command okay.
150 Opening ASCII NO-PRINT mode data connection for ls -l.
test.txt
226 Transfer finished successfully. Data connection closed.
FTP: 10 byte(s) received in 0.160 second(s) 62.50byte(s)/sec.
```

可以观察到，公网用户可以成功登录公司内的私网 FTP 服务器。

思考

什么情况下需要使用到 NAT 的双向转换？

附录
命令索引



基础配置命令	
Display	查看系统当前日期和时钟
clock timezone *time-zone-name* { **add** | **minus** } *offset*	用来对本地时区信息进行设置。*time-zone-name*指定时区名称；**add** 与通用协调时间UTC相比，*time-zone-name*增加的时间偏移量，即在系统默认的UTC时区的基础上，加上*offset*，就可以得到*time-zone-name*所标识的时区时间；**minus**与UTC时间相比，*time-zone-name*减少的时间偏移量，即在系统默认的UTC时区的基础上，减去*offset*，就可以得到*time-zone-name*所标识的时区时间；*offset*指定与UTC的时间差
display clock	用来查看系统当前日期和时钟
display users	用来查看每个用户界面的用户登录信息
display version	用来查看版本信息
quit	用来从当前视图退回到较低级别视图，如果是用户视图，则退出系统
return	用来从除用户视图外的其它视图退回到用户视图
sysname	用来设置路由器的主机名
system-view	用来使用户从用户视图进入系统视图
user privilege level *level*	用来配置用户级别。**level** *level* 指定用户级别
user-interface [*ui-type*] *first-ui-number* [*last-ui-number*]	用来进入一个或多个用户界面视图。*ui-type*指定用户界面（User-interface）的类型；*first-ui-number*指定配置的第一个用户界面编号；*last-ui-number*指定配置的最后一个用户界面编号。选择此参数，将同时进入多个用户界面视图。此参数只有在*ui-type*取值是VTY或TTY类型时才有效。*last-ui-number*的取值要比*first-ui-number*取值大
cd [.. | *directory*]	用来进入用户某个目录操作文件。修改用户当前界面的工作路径为此目录，用户当前目录为设置好的默认工作路径（如“flash:/”）。默认情况下的工作路径为flash:/；*directory*指定要进入的目标目录名称
copy *source-filename destination-filename*	用来拷贝当前设备上的某个文件到目标指定路径下。*source-filename*指定要拷贝的源文件名，字符串形式，格式为[path][file-name]；*destination-filename*指定要拷贝到的目标文件名，字符串形式，格式为[path][file-name]
delete	用来删除存储设备中的指定文件。支持" * "通配符
dir [*remote-filename*] [*local-filename*]	显示目录下的所有文件或查询文件。*remote-filename*指定查询的文件名；*local-filename*指定保存的本地文件名
ftp [[**-a** *source-ip-address* | **-I** *interface-type interface-number*] *host* [*port-number*] [**public-net** | **vpn-instance** *vpn-instance-name*]]	**-a** *source-ip-address*指定本端设备的IPv4地址；**-i** *interface-type interface-number*指定本端设备的出接口；*host*指定远程FTP server的IP地址或主机名称；*port-number*指定远程FTP服务器的端口号；**public-net**指定在公网中连接FTP服务器；**vpn-instance** *vpn-instance-name*指定远程FTP服务器端的VPN实例名
get *remote-filename* [*local-filename*]	用来下载远程文件并存储在本地。*remote-filename*指定远程FTP server上的文件名；*local-filename*指定本地文件名

（续表）

基础配置命令	
ls [*remote-filename*] [*local-filename*]	查询指定的文件。*remote-filename*指定查询的远程文件；*local-filename*指定保存的本地文件名
open [**-a** *source-ip-address* | **-i** *interface-type interface-number*] *host* [*port-number*] [**public-net** | **vpn-instance** *vpn-instance-name*]	用来与远程FTP server建立控制连接。命令中VPN实例名，标识FTP到哪一个VPN实例中的FTP server。**-a** *source-ip-address*指定源IP地址；**-i** *interface-type interface-number*指定本端可以连接出去的源接口；*host*指定远程FTP server的IP地址或主机名；*port-number*指定远程FTP server的端口号；**public-net**指定在公网中连接；**vpn-instance** *vpn-instance-name*指定服务器端的VPN实例名
mkdir *remote-directory*	用来在远程FTP服务器建立目录。*remote-directory*指定目录名
put *local-filename* [*remote-filename*]	用来将本地的文件上传到远程FTP server。*local-filename*指定本地的文件名；*remote-filename*指定远程FTP server上的文件名
telnet [**vpn-instance** *vpn-instance-name*] [**-a** *source-ip-address*] *host-name* [*port-number*]	用来从当前设备使用Telnet协议登录其他设备。**vpn-instance** *vpn-instance-name*指定通过Telnet协议登录的设备所属的VPN实例名；**-a** *source-ip-address*指定本端设备的IPv4地址；*host-name*指定远端设备的IPv4地址或主机名；*port-number*指定远端设备提供Telnet服务的TCP端口号
display ssh server status	用来查看当前SSH服务器的工作状态。当需要查询当前SSH服务器的协议版本、认证超时时间、密钥更新周期和认证重试次数时，使用此命令
display ssh user-information *username*	用来查看SSH用户信息。当需要查询SSH用户的用户名、认证类型、密钥名和服务类型时，使用此命令。*username*指定用户名
display ssh server session	用来查看当前SSH服务器的会话状态
display rsa local-key-pair public	用来查看本地密钥对中的公钥部分信息。当需要获取服务器端的公钥，并发送给客户端保存，或确认主机密钥对和服务密钥对是否存在时，使用此命令
display rsa peer-public-key [**brief** | **name** *key-name*]	用来查看SSH用户的密钥。当SSH用户配置密钥后需要查看SSH用户密钥的名称和数据时，使用此命令。**Brief**显示所有远端公钥的简明信息；**name** *key-name*指定远端公钥的名字
rsa local-key-pair create	用来创建SSH服务所需的主机密钥对和服务密钥对
sftp server enable	用来启用SSH服务器端的SFTP服务
stelnet server enable	用来启用SSH服务器端的STelnet服务
接口管理命令	
description *description*	用来设置接口描述信息
display interface [*interface-type* [*interface-number*] | **slot** *slot-id*]	用来查看接口当前运行状态和接口统计信息。*interface-type* [*interface-number*] 显示指定接口类型和接口编号的接口当前运行状态和统计信息；**slot** *slot-id* 显示指定接口板编号的接口当前运行状态和统计信息
display interface brief	用来查看接口状态和配置的简要信息
display ip interface	用来查看接口与IP相关的配置和统计信息，包括接口接收和发送的报文数、字节数和组播报文数，以及接口接收、发送、转发和丢弃的广播报文数

（续表）

接口管理命令	
display ip interface brief	用来查看接口与IP相关的简要信息，包括IP地址、子网掩码、物理链路和协议的Up/Down状态以及处于不同状态的接口数目
display this interface	用来显示当前接口视图下的接口信息
interface *interface-type interface-number*	用来进入已经存在的接口，或创建并进入逻辑接口。*interface-type interface-number*指定接口类型和接口编号。接口类型和接口编号之间可以输入空格也可以不输入空格。输入接口类型时支持用前几个字母代表整个接口名称，前提是这几个字母可以唯一标示出该接口名。比如输入"ethe"可以唯一代表ethernet。如无特殊说明，本文档所有举例只列举接口类型和接口编号之间有空格，且接口类型用全称的情况
shutdown	用来关闭当前接口
undo shutdown	用来开启当前接口
auto duplex	用来配置接口自协商模式下的双工模式
undo auto duplex	用来恢复接口自协商模式下的双工模式为默认值。当以太网接口工作在自协商模式时，默认情况下，它的双工模式是和对端接口协商得到的
auto speed	用来配置接口自协商模式下的协商速率
undo auto speed	用来恢复接口自协商模式下的协商速率为默认值。默认情况下，以太网接口自协商速率范围为接口支持的所有速率
speed	用来配置以太网接口的速率。接口工作于非自协商模式时，默认情况下，它的速率为100Mbit/s
arp broadcast enable	用来启用终结子接口的ARP广播功能
control-vid*vid* { **dot1q-termination** | **qinq-termination** }	用来指定控制VLAN和终结子接口的对应关系，从而区分同一主接口下不同终结子接口。*Vid*指定子接口控制VLAN ID，用于标识不同子接口；**dot1q-termination** 配置子接口为dot1q封装方式，适用于对带有一层Tag报文终结； **qinq-termination**配置子接口为QinQ封装方式，适用于对带有两层Tag报文终结
dot1q termination vid *vid*	用来配置子接口对一层Tag报文的终结功能。*Vid* 指定用户报文中Tag的取值
interface *interface-type interface-number.subinterface-number* [**p2mp** | **p2p**]	用来创建接口链路层封装协议为帧中继的子接口。**interface** *interface-type interface-number* 指定链路层封装协议为帧中继的接口编号；*subinterface-number* 指定子接口编号；**p2mp**配置子接口类型为点到多点，当接口封装的协议类型是帧中继时，默认的子接口类型是p2mp；**p2p**配置子接口类型为点到点
interface loopback *loopback-number*	用来创建Loopback接口。*loopback-number*指定Loopback接口的编号
链路聚合配置命令	
display eth-trunk [*trunk-id* [**interface** *interface-type interface-number* | **verbose**]]	用来查看Eth-Trunk接口的配置信息。*trunk-id*指定Eth-Trunk接口的编号；**interface** *interface-type interface-number*指定Eth-Trunk中的成员接口；**verbose** 查看Eth-Trunk接口的详细配置信息

（续表）

链路聚合配置命令	
display inter face eth-trunk [*trunk-id* [*.subnumber*]]	用来查看Eth-Trunk接口的状态信息。*trunk-id*指定Eth-Trunk接口的编号；*subnumber*指定Eth-Trunk子接口编号
eth-trunk *trunk-id*	用来将当前接口加入到指定Eth-Trunk中。*trunk-id* 指定Eth-Trunk接口的编号
interface eth-trunk *trunk-id* [*.subnumber*]	用来创建Eth-Trunk接口并进入Eth-Trunk接口视图。*trunk-id* 指定Eth-Trunk接口的编号；*subnumber*指定Eth-Trunk的子接口编号
VLAN配置命令	
display interface vlanif [*vlan-id*]	用来查看VLANIF接口的状态信息和基本配置信息。*vlan-id*指定VLAN编号
display vlan	用来查看所有VLAN的相关信息
interface vlanif *vlan-id*	用来创建VLANIF接口并进入VLANIF接口视图。*vlan-id*指定待创建VLANIF接口所对应的VLAN编号
port default vlan *vlan-id*	用来配置接口的默认VLAN并同时加入这个VLAN。*vlan-id* 配置默认VLAN的编号
port hybrid pvid vlan *vlan-id*	用来设置Hybrid类型接口的默认VLAN ID。*vlan-id* 指定Hybrid类型接口的默认VLAN编号
port hybrid tagged vlan { { *vlan-id*1[**to** *vlan-id*2] }&<1~10> | **all** }	*vlan-id*1 [**to** *vlan-id*2] 指定Hybrid类型接口所属的VLAN，其中*vlan-id1*表示被创建的第一个VLAN的编号；**to** *vlan-id*2表示被创建的最后一个VLAN的编号。*vlan-id*2的取值必须大于*vlan-id*1的取值，它和*vlan-id*1共同确定一个范围。如果不指定**to** *vlan-id*2参数，则只创建*vlan-id*1所指定的VLAN。**all**指定Hybrid接口所属的所有VLAN
port hybrid untagged vlan { { *vlan-id*1 [**to** *vlan-id*2] }&<1~10> | **all** }	用来配置Hybrid类型接口所属的VLAN，这些VLAN的帧以Untagged方式从该接口发送出去。*vlan-id*1 [**to** *vlan-id*2] 指定Hybrid类型接口所属的VLAN，其中*vlan-id*1表示被创建的第一个VLAN的编号；**to** *vlan-id*2表示被创建的最后一个VLAN的编号。*vlan-id*2的取值必须大于*vlan-id*1的取值，它和*vlan-id*1共同确定一个范围。如果不指定**to** *vlan-id*2参数，则只创建*vlan-id*1所指定的VLAN。**all**指定Hybrid接口所属的所有VLAN
port link-type { **access** | **hybrid** | **trunk** }	用来配置接口的链路类型。**access**配置接口的链路类型为access；**hybrid**配置接口的链路类型为Hybrid；**trunk**配置接口的链路类型为Trunk
port trunk allow-pass vlan { { *vlan-id*1 [**to** *vlan-id*2] }&<1~10> | **all** }	用来配置Trunk类型接口加入的VLAN。*vlan-id*1 [**to** *vlan-id*2] 指定Trunk类型接口所属的VLAN，其中*vlan-id*1表示被创建的第一个VLAN的编号；**to** *vlan-id*2表示被创建的最后一个VLAN的编号。*vlan-id*2的取值必须大于*vlan-id*1的取值，它和*vlan-id*1共同确定一个范围。如果不指定**to** *vlan-id*2参数，则只创建*vlan-id*1所指定的VLAN。**all**指定Trunk接口属于所有VLAN
port trunk pvid vlan *vlan-id*	用来设置Trunk类型接口的默认VLAN ID。*vlan-id*指定Trunk类型接口的默认VLAN编号
vlan *vlan-id*	用来创建VLAN并进入VLAN视图。*vlan-id*配置VLAN的编号
vlan batch { *vlan-id*1 [**to** *vlan-id*2] }&<1~10>	用来创建一个或批量创建多个VLAN。*vlan-id*1 [**to** *vlan-id*2] 指定VLAN的编号，其中*vlan-id*1表示被创建的第一个VLAN的编号；**to** *vlan-id*2表示被创建的最后一个VLAN的编号。*vlan-id*2的取值必须大于*vlan-id*1的取值，它和*vlan-id*1共同确定一个范围。如果不指定**to** *vlan-id*2参数，则只创建*vlan-id*1所指定的VLAN

（续表）

广域网配置命令	
display fr interface [*interface-type interface-number*]	用来查看使用帧中继协议的接口状态，包括接口是作为DTE还是DCE，以及物理状态和链路层协议状态，对于子接口，将显示子接口类型和链路层协议状态。*interface-type*指定接口类型，不指定口类型则显示所有帧中继接口的信息；*interface-number*指定接口编号
display fr map-info	用来显示帧中继地址映射表
display fr pvc-info	用来查看帧中继虚电路的配置情况和统计信息
fr dlci *dlci*	用来为帧中继接口配置虚电路并进入虚电路视图。*dlci*为帧中继接口分配的虚电路号。执行**undo fr dlci**命令时，如果不指定DLCI，则删除本接口上所有的DLCI；如果是在帧中继主接口下执行此命令，不会删除帧中继子接口下的DLCI
fr inarp [**ip** [*dlci-number* | **pvc-group** *pvc-group-name*]]	用来允许帧中继逆向地址解析功能。**ip**表示对IP网络协议进行逆向地址解析；*dlci-number*指定本地虚电路号；**pvc-group** *pvc-group-name*指定PVC组名
fr map ip { *destination-address* [*mask*] | **default** } { *dlci-number* | **pvc-group** *pvc-group-name* } [[**ietf** | **nonstandard**] [**broadcast**]]	用来增加一条帧中继地址和一个DLCI的静态地址映射。*destination-address* [*mask*]指定目的IP地址和子网掩码；**default**指定一条默认映射；*dlci-number* 指定本地的数据链路标识符；**ietf**指定帧中继接口报文格式为IETF；**nonstandard**指定帧中继接口报文格式为非标准格式；**broadcast** 指定广播方式，表示该映射上可以发送广播报文。**pvc-group** *pvc-group-name*指定PVC组名。在ATM接口上保持唯一，并且不能是合法的VPI/VCI值对，如“1/20”就不允许作为PVC名
link-protocol fr [**ietf** | **nonstandard**]	用来指定接口链路层协议为帧中继协议的封装格式。**ietf**表示IETF标准封装，按照RFC 1490规定的格式进行封装，为默认封装格式；**Nonstandard**表示非标准兼容的封装格式
link-protocol ppp	用来配置接口封装的链路层协议为PPP
ppp authentication-mode { **chap** | **pap** } [[**call-in**] **domain** *domain-name*]	用来设置本端PPP协议对对端设备的认证方式。**chap**表示采用CHAP认证方式；**pap**表示采用PAP认证方式；**call-in**表示只在远端用户呼入时才认证对方；**domain** *domain-name*表示用户认证采用的域名
ppp chap password { **cipher** | **simple** } *password*	用来配置CHAP验证的口令。**cipher**表示密码为密文显示；**simple**表示密码为明文显示；*password*设置CHAP认证的口令
ppp chap user *username*	用来配置CHAP验证的用户名。*username*设置CHAP验证的用户名，该用户名是发送到对端设备进行CHAP验证时使用的用户名
link-protocol hdlc	用来配置接口封装HDLC协议
静态路由、RIP路由协议配置命令	
Router-ID *router-id*	用来设置路由管理中的Router-ID。*router-id*指定IPv4地址形式的Router-ID
ip route-static	用来配置IPv4单播静态路由
default-cost *cost*	用来设置引入路由的路由开销值。*cost*指定所要设定的默认路由权值

（续表）

静态路由、RIP路由协议配置命令	
default-route originate [**match default** [**avoid-learning**]] [**cost** *cost*]	用来在当前路由器生成一条默认路由或者将路由表中存在的默认路由发送给邻居。**match default**表示如果在路由表中存在其他路由协议或其他RIP进程生成的默认路由，则向邻居发布该默认路由；**avoid-learning**表示避免RIP进程引入默认路由。如果路由表中已存在的默认路由为活跃状态，选用该参数可以将此路由置为不活跃状态。**cost** *cost*指定生成默认路由的度量值
display rip	用来显示RIP进程的当前运行状态及配置信息
display rip *process-id* **database** [**verbose**]	用来查看RIP发布数据库的所有激活路由。这些路由以常规RIP更新报文的形式发送。*process-id*指定RIP进程号；**verbose**显示RIP发布数据库中路由的详细信息
display rip *process-id* **interface** [*interface-type interface-number*] [**verbose**]	用来显示RIP的接口信息。*process-id*指定RIP进程号；*interface-type*指定接口类型，如GigabitEthernet、Loopback等；*interface-number*指定接口号；**verbose**指定查看RIP接口的详细信息
display rip *process-id* **route**	用来显示所有从其它路由器学来的RIP路由信息，以及与每条路由相关的不同定时器的值。*process-id*指定RIP进程号
import-route	用来从其他路由协议引入路由
network *network-address*	用来对指定网段接口使能RIP功能。*network-address*指定启用RIP的网络地址，必须是自然网段的地址
peer *ip-address*	用来指定NBMA网络中RIP邻居路由器的IP地址。配置此命令后，更新报文以单播形式发送到对端，而不采用正常的组播或广播的形式。*ip-address*指定邻居路由器的IP地址
preference { *preference* | **route-policy** *route-policy-name* }*	用来指定RIP路由的优先级，并且通过应用路由策略可以对特定的路由设置优先级。*preference*指定路由的优先级；**route-policy**表示应用路由策略，对满足条件的特定路由设置优先级；*route-policy-name*指定路由策略名称
rip	用来启用系统视图下的指定RIP进程
rip authentication-mode	用来配置RIP-2的验证方式及验证参数。每次验证只支持一个验证字，新输入的验证字将覆盖旧验证字
rip input	用来允许指定接口接收RIP报文
rip metricin *value*	用来设置接口接收RIP报文时给路由增加的度量值。*value*指定对接收到的路由增加度量值
rip metricout	用来设置接口发送RIP报文给路由增加的度量值
rip output	用来允许接口发送RIP报文
rip poison-reverse	用来配置RIP更新报文的毒性反转进程
rip split-horizon	用来配置RIP水平分割功能
rip summary-address *ip-address mask* [**avoid-feedback**]	用来设置一个RIP路由器发布一个聚合的本地IP地址。*ip-address*表示需要聚合的网络IP地址；*mask*表示网络掩码；**avoid-feedback**表示禁止从此接口学习到相同的聚合路由
rip version { **1** | **2** [**broadcast** | **multicast**] }	用来设置接口的RIP版本。**1**表示RIP-1报文；**2**表示RIP-2报文；**broadcast**表示以广播方式发送RIP-2报文；**multicast**表示以组播方式发送RIP-2报文。默认情况下，RIP-2报文使用组播方式发送

（续表）

静态路由、RIP路由协议配置命令	
silent-interface { **all** | *interface-type interface-number* }	用来抑制RIP接口，使其只接收报文用来更新自己的路由表，而不发送更新报文。**all**指定抑制所有RIP接口；*interface-type interface-number* 指定接口类型及编号
summary	用来启用RIP有类聚合。聚合后的路由以使用自然掩码的路由形式发布
timers rip *update age garbage-collect*	用来调整定时器。*update*指定路由更新报文的发送间隔；*age*指定路由老化时间；*garbage-collect*指定路由被从路由表中删除的时间（标准中定义的garbage收集时间）
version { **1** | **2** }	用来指定一个全局RIP版本；**1**指定RIP-1版本；**2**指定RIP-2版本
OSPF配置命令	
abr-summary *ip-address mask* [[**advertise** | **not-advertise**] | **cost** *cost*] *	用来在区域边界路由器（ABR）上配置路由聚合。*ip-address*指定IP地址，点分十进制形式；*mask*指定IP地址的掩码，点分十进制形式；**advertise** | **not-advertise**指定是否发布这条聚合路由，默认为发布聚合路由；**cost** *cost* 设置聚合路由的开销。当此参数不配置时，则取所有被聚合的路由中最大的那个开销值作为聚合路由的开销
area *area-id*	用来创建并进入OSPF区域视图。*area-id*指定区域的标识，可以是十进制整数或IP地址格式
asbr-summary *ip-address mask* [**not-advertise** | **tag** *tag* | **cost** *cost* | **distribute-delay** *interval*] *	用来设置OSPF对引入的路由进行聚合。*ip-address*指定IP地址，点分十进制格式；*mask*指定掩码，点分十进制格式；**not-advertise**不通告聚合路由，如果不指定该参数则将通告聚合路由；**tag** *tag* 用于通过路由策略控制路由发布；**cost** *cost*设置聚合路由的开销。当此参数不配置时，对于Type1类外部路由，取所有被聚合路由中的最大开销值作为聚合路由的开销；对于Type2类外部路由，则取所有被聚合路由中的最大开销值再加上1作为聚合路由的开销。 **distribute-delay** *interval* 指定延迟发布聚合路由的时间
authentication-mode	用来指定OSPF区域所使用的验证模式及验证口令。配置该命令相当于在指定区域所有路由器接口下使用相同的验证
display ospf interface	用来显示OSPF的接口信息
display ospf lsdb	用来显示OSPF的链路状态数据库信息
display ospf peer	用来显示OSPF中各区域邻居的信息
display ospf routing	用来显示OSPF路由表的信息
network *address wildcard-mask*	用来指定运行OSPF协议的接口和接口所属的区域。*address*指定接口所在的网段地址；*wildcard-mask*指定掩码的反码，相当于将IP地址的掩码反转（0变1，1变0）。其中，“1”表示忽略IP地址中对应的位，“0”表示必须保留此位
ospf	用来创建并运行OSPF进程
ospf authentication-mode	用来设置相邻路由器之间的验证模式及验证字
ospf cost *cost*	用来配置接口上运行OSPF协议所需的开销。*cost*指定运行OSPF协议所需的开销
ospf timer dead *interval*	用来设置OSPF的邻居失效时间

（续表）

OSPF配置命令	
ospf timer hello *interval*	用来设置接口发送Hello报文的时间间隔
peer *ip-address* [**dr-priority** *priority*]	用来配置NBMA网络上相邻路由器IP地址及DR优先级。*ip-address*指定邻接点的IP地址；**dr-priority** *priority*指定表示网络邻居的优先级的相应数值
VRRP配置命令	
display vrrp	用来查看当前VRRP备份组的状态信息和配置参数
vrrp vrid authentication-mode	用来设置VRRP备份组的认证字
vrrp vrid *virtual-router-id* **preempt-mode disable**	用来设置备份组中路由器采用非抢占方式。**vrid** *virtual-router-id* 指定VRRP备份组号
vrrp vrid*virtual-router-id* **priority** *priority-value*	用来设置路由器在备份组中的优先级。**vrid** *virtual-router-id*指定VRRP备份组号；**priority** *priority-value* 优先级取值
vrrp vrid *virtual-router-id* **virtual-ip** *virtual-address*	用来创建VRRP备份组并为备份组指定虚拟IP地址。**vrid** *virtual-router-id* 指定VRRP备份组号；**virtual-ip** *virtual-address*指定虚拟IP地址
vrrp virtual-ip ping enable	用来启用Master设备响应ping报文
生成树配置命令	
active region-configuration	用来激活MST域配置，包括域名、修订级别、VLAN和MSTI的映射关系
display stp	用来显示生成树实例的状态信息和统计信息
instance	用来将指定VLAN映射到指定MSTI上
region-name *name*	用来配置MST域名
revision-level *level*	配置MSTP修订级别
stp { **disable** | **enable** }	指定MSTP功能的状态。**disable**表示禁用MSTP功能；**enable**表示启用MSTP功能
stp cost	用来配置当前接口在指定MSTI上的接口路径开销
stp edged-port { **disable** | **enable** }	指定当前接口配置为边缘接口或非边缘接口。默认情况下，所有接口均为非边缘接口。**disable**表示配置当前的接口为非边缘接口；**enable**表示配置当前的接口为边缘接口
stp mode { **mstp** | **stp** | **rstp** }	用来配置设备的MSTP工作模式
stp [**instance** *instance-id*] **priority** *priority*	用来配置在指定MSTI中的优先级。**instance** *instance-id*指定MSTI，其中*instance-id*表示MSTI的编号。*priority* 指定优先级，优先级值越小，则优先级越高
stp region-configuration	用来进入MST域视图
stp [**instance** *instance-id*] **root primary**	用来配置当前交换机作为指定MSTI的根交换机

（续表）

生成树配置命令	
stp [**instance** *instance-id*] **root secondary**	用来配置当前交换机作为指定MSTI的备份根交换机。**instance** *instance-id*指定MSTI，其中*instance-id*表示MSTI的编号
stp root-protection	用来启动当前接口的Root保护功能
stp tc-protection	用来启用交换机对TC类型BPDU报文的保护功能
stp timer hello *hello-time*	用来配置Hello Time时间
stp timer max-age *max-age*	用来配置Max Age时间，即接口上的BPDU老化时间
stp timer-factor *timer-factor*	通过设定Hello Time的倍数（Timer Factor）来配置交换机的超时时间
ARP、DHCP配置命令	
arp-proxy enable	用来启动接口的路由式Proxy ARP功能
arp static	用来配置静态ARP映射表
display arp all	用来查看所有接口板上非重复的ARP映射表和统计信息
display arp static	用来查看所有接口板的静态ARP映射表
reset arp	用来清除ARP映射表中的ARP项
dhcp enable	用来启用DHCP功能
dhcp relay release	用来通过DHCP中继向DHCP服务器发出释放客户端申请到的IP地址的请求
dhcp relay server-ip *ip-address*	用来配置DHCP中继所代理的DHCP服务器地址
dhcp relay server-select *group-name*	用来配置DHCP中继所对应的DHCP服务器组
dhcp select global	用来启用接口的DHCP服务功能，指定S5700从全局地址池分配地址
dhcp select interface	用来启用接口的DHCP服务功能，同时根据接口地址创建接口地址池，指定交换机从接口地址池分配地址
dhcp select relay	用来启用DHCP Relay功能。启用DHCP Relay功能后，VLANIF接口对接收到的DHCP报文通过中继发送到外部DHCP Server，由外部DHCP Server分配地址
dhcp-server *ip-address* [*ip-address-index*]	用来在DHCP服务器组中添加DHCP服务器。*ip-address*指定DHCP服务器IP地址；*ip-address-index*指定IP地址的索引。配置DHCP服务器IP地址时，可以选择指定IP地址索引*ip-address-index*，如不指定索引，系统将自动分配一个空闲的索引，配置的20个IP地址和地址索引一一对应。删除DHCP服务器地址时，可以指定IP地址或地址索引进行删除
dhcp server dns-list	用来指定VLANIF接口地址池下的DNS服务器
dhcp server (domain-name)	用来指定DHCP服务器接口地址池的域名
dhcp server exclude-ip-address *start-ip-address* [*end-ip-address*]	用来配置VLANIF接口地址池中不参与自动分配的IP地址范围。*start-ip-address*指定不参与自动分配的IP地址段的起始IP地址；*end-ip-address*指定不参与自动分配的IP地址段的结束IP地址

（续表）

ARP、DHCP配置命令	
dhcp server group *group-name*	用来创建一个DHCP服务器组，并进入DHCP服务器组视图
dhcp server lease	用来配置DHCP服务器VLANIF接口地址池中IP地址的租用有效期限
display dhcp relay	用来查看VLANIF接口对应的DHCP服务器组的信息
display ip pool	用来查看设备上已经配置的IP地址池信息
dns-list	用来为全局地址池配置DNS服务器地址
excluded-ip-address	用来配置IP地址池中不参与自动分配的IP地址范围
gateway *ip-address*	用于在DHCP服务器组视图下配置DHCP中继向DHCP服务器的出口网关地址
gateway-list *ip-address* &<1~8>	用来配置DHCP服务器全局地址池的出口网关地址
ip pool *ip-pool-name*	用来创建全局地址池
lease	用来配置DHCP全局地址池下的地址租期
network *ip-address* [**mask** { *mask* | *mask-length* }]	用来配置全局地址池下可分配的网段地址
IPv6配置命令	
display ipv6 interface [*interface-type interface-number* | **brief**]	用来查看接口的IPv6信息。*interface-type* 指定接口类型；*interface-number* 指定接口编号；**brief**显示接口的摘要信息
ipv6	用来启用设备转发IPv6单播报文，包括本地IPv6报文的发送与接收
ipv6 address { *ipv6-address prefix-length* | *ipv6-address/prefix-length* }	用来手工配置接口的站点本地地址或全球单播地址。*ipv6-address* 指定IPv6地址；*prefix-length*指定前缀长度，只能在Loopback接口上配置前缀长度是128bit的IPv6地址
ipv6 address auto link-local	用来为接口配置自动生成的链路本地地址
ipv6 address { *ipv6-address prefix-length* |*ipv6-address / prefix-length* } **eui-64**	用来给接口配置EUI-64格式的站点本地地址或全球单播地址。*ipv6-address* 指定IPv6地址；*prefix-length* 指定前缀长度
ipv6 address *ipv6-address* **link-local**	用来手动配置指定接口的链路本地地址
ipv6 enable	用来在接口上启用IPv6功能
GRE配置命令	
destination *dest-ip-address*	用来对隧道指定目的端IP地址。*dest-ip-address*指定Tunnel的目的地址
display interface tunnel [*interface-number*]	用来查看Tunnel接口的信息。*interface-number*指定Tunnel接口的编号
display tunnel-info	用来查看隧道信息

（续表）

GRE配置命令	
interface tunnel *interface-number*	用来创建一个Tunnel接口，并进入该Tunnel接口视图。*interface number* Tunnel接口的编号，具体编号形式与实际使用的硬件设备相关。Tunnel接口编号只具有本地意义，隧道两端可以使用不同的接口编号
source	用来指定Tunnel接口的源地址或源接口
tunnel-protocol { **gre** | **none** }	用来配置隧道模式。**gre**表示配置Tunnel接口为GRE隧道模式，GRE隧道为点到点模式；**none**配置Tunnel接口的隧道模式为NONE，该模式的隧道不具备任何功能，为了使用该隧道接口，必须将隧道模式切换为其他模式

封面设计：Dile迪利

作为“华为ICT认证系列丛书”中一本不可多得的实验学习指南，《HCNA网络技术实验指南》是由华为技术有限公司与泰克网络实验室联合创作的一本权威性的HCNA网络技术学习用书。本书基于eNSP搭建企业网络真实场景，并给出大量配置案例，将真实场景与配置实例紧密结合，不但使读者能熟悉华为VRP命令体系，获得操作维护华为网络设备的能力，而且能帮助读者进一步深刻理解HCNA网络技术原理，具备企业网络规划、部署、故障排除的能力。

本书对HCNA网络技术中的各个知识点进行了深入剖析，并为每项知识点精心设计、量身打造了真实的应用场景，配以Step-by-Step方式的详尽步骤讲解，使其条理清晰、繁而不杂，一学即会；并且，每个实验的结尾还设计有能够深入启发的思考题，帮助读者提升对相关知识的进一步学习和理解，使其能够真正地学有所思，学有所获，学有所效。







ISBN 978-7-115-34319-2
定价：69.00 元

分类建议：计算机网络
人民邮电出版社网址：www.ptpress.com.cn